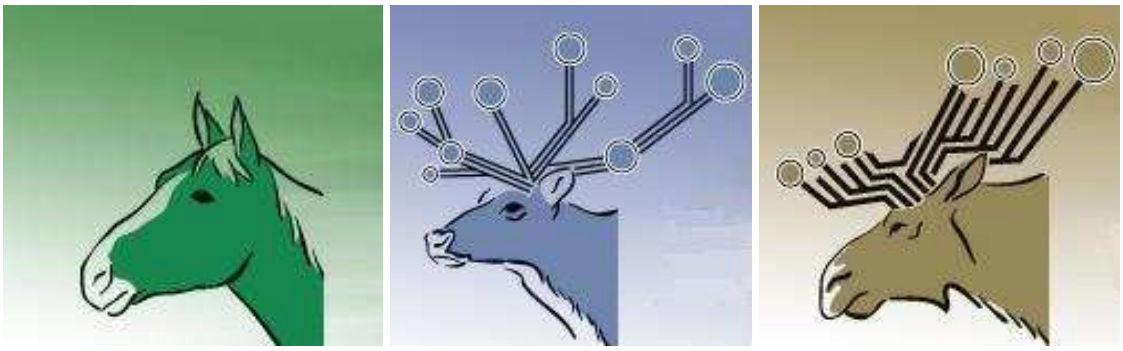
Proceedings of the 6th Bayesian Modelling Applications Workshop

How biased are our numbers?

Silja Renooij	Utrecht University
Hermi J.M. Tabachneck-Schijf	Utrecht University
Suzanne M. Mahoney	Innovative Decisions, Inc.



Held in conjunction with the 21st Annual Conference on Learning Theory (COLT 2008), the 25th International Conference on Machine Learning (ICML 2008), and the 24th Conference on Uncertainty in Artificial Intelligence (UAI 2008), Helsinki University, Helsinki, Finland.

Workshop committee:

Suzanne M. Mahoney	Innovative Decisions, Inc.	co-chair
Silja Renooij	Utrecht University	co-chair
Hermi J.M. Tabachneck-Schijf	Utrecht University	co-chair

John Mark Agosta	Intel Research
Russel Almond	Educational Testing Service
Dennis M. Buede	Innovative Decisions, Inc.
Marek Druzdzel	University of Pittsburgh
Linda van der Gaag	Utrecht University
Judy Goldsmith	University of Kentucky
Sean Guarino	Charles River Analytics
Kathryn Laskey	George Mason University
Ann Nicholson	Monash University
Jonathan Pfautz	Massachusetts Institute of Technology
Juan-Diego Zapata-Rivera	Educational Testing Service

Foreword

As in previous years, the goal of the Bayesian Modelling Applications workshop is to provide a focused, informal forum for fruitful exchanges among theorists, practitioners and tool developers. Discussions may cover research questions and insights, methodologies, techniques, and experiences with applications of Bayesian models to any particular problem domain. This year we address the special theme

How biased are our numbers ?

We have composed an interesting program of selected contributions that focus on issues relating to (probability) biases in applications of Bayesian networks. For example, in constructing a Bayesian model, the probabilistic information required for establishing its numerical parameters can be obtained from data, human experts, a mix of these, or from yet other sources, all of which are known to be biased. How can the biases in the sources of probabilistic information be identified? How can the degree of bias, and its effect on the resulting model and its behaviour, be established? Is it possible to correct for these biases? Do the dedicated elicitation techniques that are being designed for the purpose of eliciting probabilities from human experts forestall, for example, biases and over-commitments of the resulting model? Are these techniques efficient, easy to use, and scale up to building large models? In verifying the probability assessments and behaviour of the model under construction, biases in the numbers and in the interpretation of these numbers can be expected. What type of bias can be expected, and how can it be identified? What kind of probabilistic information, possibly computed from the model under construction, do you feed back to, for example, a human expert? How do you communicate such information? Is it interpreted as intended?

This year, the workshop includes a session partially dedicated to an inference evaluation, held prior to the workshops/conferences. We are looking forward to yet another fruitful edition of the Applications workshop, hoping it will provide for identifying useful insights, techniques and future challenges for all research communities concerned with reasoning under uncertainty.

Silja Renooij
Hermi J.M. Tabachneck-Schijf
Suzanne M. Mahoney

Schedule for the 6th Bayesian Modelling Applications Workshop

09:00 – 09:30 Welcome and Introduction by Silja Renooij

09:30 – 10:30 Session I: Probability elicitation and bias
Moderator: Marek Druzdzel

Observations from field trials with several elicitation techniques in an ecological domain
C.R. Thomas, A.E. Nicholson, and B.T. Hart

Relieving the elicitation burden of Bayesian Belief Networks
B.W. Wisse, S.P. van Gosliga, N.P. van Elst, and A.I. Barros

10:30 – 11:00 Coffee Break

11:00 – 12:30 Session II: Model elicitation and bias
Moderator: John-Mark Agosta

A Bayesian approach to learning in fault isolation
H. Wettig, A. Pernestål, T. Silander, and M. Nyberg

Hypothesis Management Framework: a flexible design pattern for belief networks in decision support systems
S.P. van Gosliga and I. van de Voorde

An experimental procedure for evaluating user-centered methods for rapid Bayesian network construction
M. Farry, J. Pfautz, Z. Cox, A. Bisantz, R. Stone, and E. Roth

12:30 – 14:30 Lunch Break

14:30 – 16:00 Session III: Biased inference
Moderator: Silja Renooij

The impact of overconfidence bias on practical accuracy of Bayesian network models: an empirical study
M.J. Druzdzel and A. Oniśko

Results of the probabilistic inference evaluation[1]
R. Dechter and A. Darwiche

16:00 – 16:30 Coffee Break

16:30 – 18:00 Session IV: Making bias explicit
Moderator: Finn Jensen

Methods for representing bias in Bayesian networks
E. Carlson, S. Guarino, and J. Pfautz

Discussion and Closing

[1] See next page

Evaluating Probabilistic Reasoning Systems

Adnan Darwiche and Rina Dechter

The *probabilistic reasoning evaluation* took place during the month preceding the UAI conference. Its results will be presented at the applications workshop, and a full report on the methodology, benchmarks and results will be posted at the evaluation webpage following the UAI conference.

Motivation

Over the past two decades a variety of exact and approximate algorithms were developed across several communities (e.g. UAI, NIPS, SAT/CSPs) for answering optimization and likelihood queries over probabilistic graphical models. Since all these tasks are NP-hard, theoretical guarantees are rare and empirical evaluation becomes a central evaluation tool. Yet, the empirical comparison between algorithms requires agreement on representations, benchmarks and evaluation criteria which is challenging, especially in the context of approximation algorithms.

Some communities have already addressed similar challenges through yearly empirical evaluations and competitions (e.g. SAT, CSP and planning) which proved effective, leading to algorithmic advances and to software development and dissemination. We believe that such an effort could benefit probabilistic inference algorithms as well. Probabilistic reasoning presents additional challenges, however, as it tends to be harder, requires heterogenous knowledge representation frameworks, and must deal with the issue of evaluating approximate inference algorithms.

Goals

Our goal is to use the evaluation as a process that will help establish some standards for evaluating probabilistic reasoning systems based on both exact and approximate algorithms. Another long term goal is to reinforce a tradition of building and sharing probabilistic reasoning systems that allow easy access to state-of-the-art inference algorithms by members of the broader scientific and engineering communities. We hope to achieve a number of objectives:

- Increase the utilization of probabilistic inference algorithms in real-world applications by reducing the investment needed for building applications based on probabilistic reasoning.
- Allow newer members of the inference community to quickly capitalize on the expertise of more senior members of the community by providing broader access to existing code.
- Foster an environment where reported empirical results are accompanied by the very systems used to obtain them.

The actual UAI'08 *probabilistic reasoning evaluation* took place during the month preceding the conference and its results are presented and discussed during the applications workshop. The evaluation includes both Bayesian and Markov networks and consider three inference tasks: probability of evidence (partition function), most probable explanations (also called MPE or energy minimization), and node marginals. The evaluation will consider both exact and approximate algorithms, especially anytime algorithms that improve their approximations with time. Details of the evaluation can be found at:

`http://graphmod.ics.uci.edu/uai08/Evaluation`

A full report on the methodology, benchmarks and the results will be posted at the evaluation webpage following the conference.

Organizing Committee

- Fahiem Bacchus: http://www.cs.toronto.edu/∼fbacchus/
- Jeff Bilmes: http://ssli.ee.washington.edu/people/bilmes/
- (co-chair) Adnan Darwiche: http://www.cs.ucla.edu/∼darwiche/
- (co-chair) Rina Dechter: http://www.ics.uci.edu/∼dechter/
- Hector Geffner: http://www.tecn.upf.es/∼hgeffner/
- Alexander Ihler: http://www.ics.uci.edu/∼ihler/
- Joris Mooij: http://www.jorismooij.nl/
- Kevin Murphy: http://www.cs.ubc.ca/∼murphyk/

Papers

Observations from field trials with several elicitation techniques in an ecological domain

Colette R. Thomas*
Water Studies Centre
Monash University,
VIC, 3800, Australia

Ann E. Nicholson
Faculty of Information Technology
Monash University
VIC, 3800, Australia

Barry T. Hart
Water Studies Centre
Monash University,
VIC, 3800, Australia

Abstract

Quantitative ecologists use Bayesian networks (BNs) to integrate their collective understanding of system processes, and to adaptively investigate management alternatives. Consequently, subjective probability assessments are often critical for ecological BNs. Several published probability elicitation techniques were trialled in development of a prototype ecological BN. These included verbal, numeric, text and matrix formats. Observations of the participant's preferences for and performances under the different formats are described and discussed.

1 INTRODUCTION

We wanted to construct a BN collaboratively with the key end-user group for the domain, namely tropical seagrass managers and scientists in the Great Barrier Reef World Heritage Area (GBRWHA), in northeastern Australia. In this region elevated nutrient and sediments entering the GBRWHA from river flows are considered one of the most important land-based influences on the system (Brodie et al. 2007), although the issue has been contentious (e.g. Starck 2005). A risk-based approach using BNs was considered way to tackle these problems in the GBRWHA, however data scarcity meant that experts were required to provide some of the probability estimations for the BN.

Given the complexity and data scarcity of most ecological systems, significant effort is required to maximise the extraction of information from available data. Most data types can be adapted to BN analysis this is one of the reasons why BNs are so appealing to ecological risk practitioners. However, rarely in an ecological application are all pertinent relationships represented adequately, if at all, by empirical data. In these instances machine learning and expert knowledge can be used to quantify these system relationships with probabilities. Many elicitation methods are available, but little guidance exists about how to choose between them or the biases they may introduce. We used our need for expert probabilities as an opportunity to informally trial several extant techniques. After introducing our domain, we describe the methods used, and our observations of participant responses.

2 THE ECOLOGICAL DOMAIN

The effect of land-based activities on marine ecosystems is a matter of global concern (GESAMP 2001). With the recognition of these persistent problems also comes acknowledgement that they cannot be properly managed without understanding the interdependencies that exist between marine and land-based systems (GESAMP 2001). This is equally true for coastal lands draining to the GBRWHA, which extends 2,000 km along the coast (Brodie et al. 2001a). The GBRWHA contains approximately 3,000 reefs, large areas of seagrass and inshore mangrove forests (Brodie et al. 2001a).

The region shown in Figure 1 is primarily agricultural, covering approximately 410,000 km^2 of land (Rayment 2005) draining directly into the Great Barrier Reef lagoon. Agricultural runoff containing soil, nutrients and chemicals drains from catchments into rivers which discharge into the GBRWHA. Elevated turbidity and nutrients levels have been measured in river plumes extending from many river mouths into the lagoon (Devlin et al. 2001b, Brodie et al. 2001b, Furnas 2003), however direct linkages between river water quality and the health of GBR ecosystems remain difficult to establish (Crossland et al. 1997).

Seagrasses are among the most productive ecosystems in the world (Duarte & Chiscano 1999). The global

*Current address: CSIRO Sustainable Ecosystems Davies Laboratory, Townsville, QLD, 4814, Australia.

Figure 1: Catchments of the Great Barrier Reef World Heritage Area, indicating the study catchment.

ecosystem services provided by seagrasses have been valued at US$3.8 trillion per year (Costanza et al. 1997). Seagrasses provide connectivity between mangroves and reefs (Mumby et al. 2004), habitat and nursery areas for algae, invertebrates and fish (Heck Jr. et al. 2003), and are the primary food source of sea turtles and dugong (Marsh et al. 1999, Aragones et al. 2006). Dugong and sea turtles are vulnerable to extinction globally (IUCN 2000) and their protection in the GBRWHA is a condition that must be met to maintain World Heritage listing.

Threatened species can be conserved if the ecosystems they use for food and shelter are protected. Ecological risk analysis can help identify the biophysical factors and processes that maintain or threaten the health of these ecosystems (Hart et al. 2006). However, ecological knowledge is notoriously insufficient for most ecological risk analysis applications. This is particularly true in Australia, where landscapes are vast relative to the resources available to observe them. Subjective probability assessments are a critical data source to fill these gaps.

3 PREPARATION

The difficulties of BN graph-building in the absence of substantial practical guidance has been acknowledged in the literature (Neil et al. 2000). However, valuable contributions to the development and communication of a coherent ecological BN methodology are increasing (e.g. Cain 2001, Ticehurst et al. 2007). In particular the Quantitative Knowledge Engineering of Bayesian Networks (Q-KEBN) methodology (Woodberry et al. 2004, Pollino et al. 2005) provides a broad framework for parameterising and evaluating BNs. Recent research has seen the development of a new framework for structural elicitation, and the extension of the parameter estimation and evaluation phases of the Q-KEBN method (Thomas et al. 2005, Thomas 2008).

The new framework was applied as follows. A tiered bottom-up approach was used to simplify a complex descriptive model to a smaller, more focused model of roughly half the size. The process worked through a rough hierarchy of system specificity (primary, secondary and tertiary factors controlling seagrass ecology) to create a graphical model of the system. Graphical modelling was followed by a phase of explicit simplification, then a phase of critical review and verification. The simplified model provided a starting point for parameterisation and refinement tasks. Automated methods were not used to learn the network structure. Once the qualitative structural characteristics were identified, relationships were quantified and parameterised (Thomas 2008).

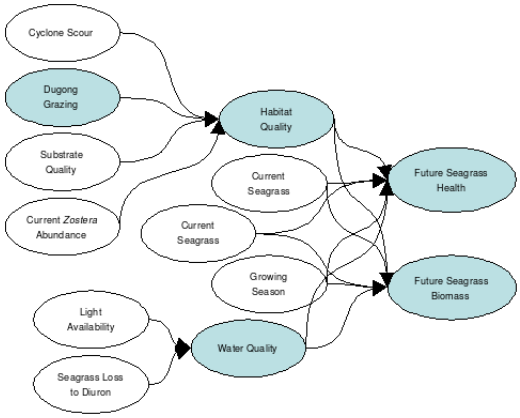


Figure 2: Seagrass health and abundance submodel

Six seagrass experts were invited to provide subjective probabilities over nodes relating directly to their area of expertise (seagrass ecology). These experts had participated in structural elicitation workshops and were familiar with the BN domain. Details of the interviewing process and examples of the Verbal Elicitor software (Hope et al. 2002) and probability assess-

ment worksheets supplemented the invitation to participate, and three experts accepted and participated in the interviews. Research shows that three to five good quality experts are often sufficient for similar, forecasting, tasks (Clemen & Winkler 1999).

Research also shows that if experts are made aware of potential biases, and are provided with training and feedback, the incidence of bias is likely to be reduced (Kahneman et al. 1982, Merkhofer 1987, List 2001). Accordingly, seagrass experts were provided with background material describing how heuristics and biases can influence subjective judgment. Materials were also provided that described and placed nodes in the context of the wider BN, and explained concepts of causal interaction and independence that were relevant to later CPT partitioning tasks. Experts were allowed approximately two weeks to digest and, if required, clarify the material before committing themselves to the elicitation process. Experts were interviewed once, individually, in private meeting rooms at or near their workplaces. All experts were interviewed by the same person.

Training sessions were used at the beginning of each interview to familiarise experts with BN concepts and allow them to experiment with all response formats. Training began with a brief explanation of BN concepts and components. The Animals BN (Norsys 2007) and a domain-relevant BN called Simple Eutrophication (Webb unpubl.) were used to demonstrate how BNs work. The Animals BN is a simple, qualitative school-level animal classification network and the Simple Eutrophication BN is a scientific algal bloom generation network – a context familiar to the experts. Experts used these BNs to test run all formats except the freehand sketch.

'Test runs' started with a two-parent node from Animals, but the CPTs became progressively more complex as the training continued, moving to the Simple Eutrophication BN. Experts were provided with an example of each elicitation format. Parent and child node details on these examples had been completed by the knowledge engineer prior to training, ensuring that all experts were trained on the same information. The response areas on the forms had been left blank. The expert was given a copy of each format and for the first test run they completed each form with as much assistance as they requested. Subsequent forms were provided for remaining examples, and the amount of assistance was progressively reduced until the experts were confident enough to use each format unassisted.

The efficacy of the training in reducing bias could not be measured for practical reasons. Similarly cost and practicality issues prevented feedback being provided to experts about the accuracy of their subjective judgments. Each interview took up to eight hours, with breaks provided every two hours.

4 TOOLS

Five nodes required subjective probabilities to be provided by experts. Three nodes (Future Seagrass Biomass, Future Seagrass Health, Dugong Grazing) had a parent node that also required subjective assessment. Probabilities for these three nodes were elicited last, ensuring that experts thoroughly understood the parent variables prior to specifying associated child node probabilities.

Experts were encouraged to complete as many probability assessments as possible. To facilitate this, the coding effort required from experts was reduced using four strategies, presented below.

1. Start with simpler nodes and work up to more complex assessments. Effort was made to simplify the range of state combinations (i.e. the size of the CPT) of the first nodes that were elicited, so experts did not become overwhelmed by the time they came to assess the two critical, and complex, endpoint nodes late in the day. To further insure that endpoint nodes received sufficient assessment, a rough guide to the amount of time that could be spent on each node was provided.

2. Reduce the number of assessments to those lying in critical areas of the distribution. This was achieved either by eliciting the best, worst and moderate cases or the 10th, 50th and 90th percentile regions of the distribution before gathering everything else in between, or by directing the experts to complete assessments for the cases they felt most confident about before contemplating more difficult assessments.

3. If it became clear that an expert could not complete the task within the session, the most difficult parent state combinations were set aside entirely and one child state was omitted from the remaining assessments. Omitted child states were later completed using a simple default rule requiring that the probabilities of the child states sum to one.

4. To provide flexibility in probability coding and response tasks, five different response formats were provided. Prior to training, each format was first explained. Training began with small and conceptually simple nodes.

   Experts could use any of the five response formats provided. The formats used were:

- graph paper for sketching probability distributions and associated parameters. Domain experts sketch the distribution they believe best represents the parent-child relationship, indicating pertinent parameter values where appropriate (e.g. mean, maxima).

- the Verbal Elicitor software (Hope et al. 2002; Figure 3). This software, based on work in van der Gaag et al. (1999), allows entry of probability values in ordinary English. The domain expert makes qualitative assessments using a scale with numerical and verbal anchors, by selecting a verbal cue such as 'unlikely' or 'almost certain'. The associated numerical probabilities are either set manually or optimised to minimise probabilistic incoherency.

- text-scale worksheet (Figure 4). This method is adapted from van der Gaag et al. (1999). The knowledge engineer reads aloud the description of the parent-child state combination. The expert circles the preferred verbal or numeric anchor, or slashes the scale axis at a preferred point along it.

- matrix worksheet (Figure 6). This method is adapted from Laskey & Mahoney (2000). Domain experts complete the full series of dependent variable state responses, given information provided on the conditioning variables. A copy of the same verbal-numeric scale used in the above format was provided to enable choice between verbal and numeric responses.

- partitioned conditional probability table matrices (Figure 5). Domain experts identified conditioning variable states that would not change the value of the dependent variable response. A copy of the same verbal-numeric scale used in the above format was provided to enable choice between verbal and numeric responses.

Experts could change response formats between nodes but were discouraged from changing from one format to another in the middle of a node assessment. Experts were given the choice of response format only for nodes with less than three parents. For nodes with more than three parents CPT partitioning was used. Experts were encouraged to use verbal-numeric responses across all formats but were never constrained to do so.

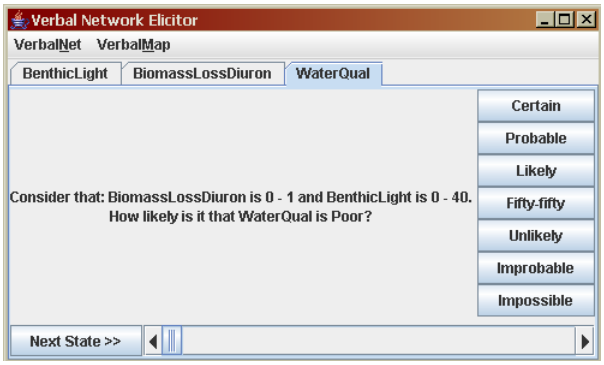


Figure 3: Screenshot from the Verbal Elicitor software (Hope et al. 2000)

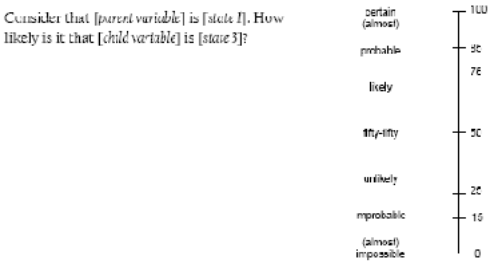


Figure 4: Extract from the text-scale worksheet (adapted from van der Gaag et al. 1999).

Cyclone Scour?	Substrate Quality	Zostera Biomass	% Lost to Dugong	Habitat Quality Poor	Moderate	Good
yes	low	0 to 5	0 to 10	probable	improbable	impossible
yes	low	0 to 5	10 to 90	probable	improbable	impossible
yes	low	0 to 5	90 to 100	probable	improbable	impossible
yes	low	5 to 15	0 to 10	probable	improbable	impossible
yes	low	5 to 15	10 to 90	probable	improbable	impossible
yes	low	5 to 15	90 to 100	probable	improbable	impossible
yes	low	15 to 100	0 to 10	probable	improbable	impossible
yes	low	15 to 100	10 to 90	probable	improbable	impossible
yes	low	15 to 100	90 to 100	probable	improbable	impossible
yes	high	0 to 5	0 to 10	probable	improbable	impossible
yes	high	0 to 5	10 to 90	probable	improbable	impossible
yes	high	0 to 5	90 to 100	probable	improbable	impossible
yes	high	5 to 15	0 to 10	probable	improbable	impossible
yes	high	5 to 15	10 to 90	probable	improbable	impossible
yes	high	5 to 15	90 to 100	probable	improbable	impossible
yes	high	15 to 100	0 to 10	probable	improbable	impossible
yes	high	15 to 100	10 to 90	probable	improbable	impossible
yes	high	15 to 100	90 to 100	probable	improbable	impossible
no	low	0 to 5	0 to 10			
no	low	0 to 5	10 to 90			
no	low	0 to 5	90 to 100			
no	low	5 to 15	0 to 10			
no	low	5 to 15	10 to 90			
no	low	5 to 15	90 to 100			
no	low	15 to 100	0 to 10			
no	low	15 to 100	10 to 90			
no	low	15 to 100	90 to 100			
no	high	0 to 5	0 to 10			
no	high	0 to 5	10 to 90			
no	high	0 to 5	90 to 100			
no	high	5 to 15	0 to 10			
no	high	5 to 15	10 to 90			
no	high	5 to 15	90 to 100			
no	high	15 to 100	0 to 10			
no	high	15 to 100	10 to 90			
no	high	15 to 100	90 to 100			

Figure 5: The CPT for the Habitat Quality node. A partition over the Cyclone Scour node is indicated with double lines

5 OBSERVATIONS ON ELICITATION PROCESSES

5.1 Biases in approach selection

Zimmer (1984) claims that different presentation modes put different emphasis on different areas of the problem-space, and Windschitl and Wells (1995) show that verbal expressions of uncertainty are more affected by presentation format than are numeric expressions. Our observations appear to support this, because in our study the response format appeared to play a role in probability elicitation results. No single format was collectively favoured by the experts over the others. Interestingly, the option to sketch the node's probability distribution was never taken up by experts during elicitation. This might indicate that familiarising experts with training materials before elicitations begin has some benefit. However, the effect of training on format preference was not tested in this study so we cannot be sure.

Text-scale worksheets and the VE software were generally preferred in both the training sessions and during elicitations of simple BN nodes. As node relationships became more complex, experts tended to prefer matrix-style formats and were eventually constrained to CPT partitioning formats for the final two, complex, nodes (Future Seagrass Biomass and Future Seagrass Health). Overall, one expert preferred the VE format and one preferred the text-scale format using verbal responses. The third expert preferred the matrix worksheet using numeric responses, stating that scientists were more accustomed to receiving information in numeric/matrix rather than verbal/text formats.

During training assessments with non-matrix formats (using VE and the text-scale worksheet) some experts showed a tendency to prefer positive cues. Answers were often bunched at the upper end of the verbal scale, with experts showing preferences for cues such as 'likely' and 'probable', and avoiding cues such as 'unlikely' or 'improbable', even though they were attempting to represent small probabilities.

The pattern was not clearly observed during assessments using probability matrix formats, indicating that the assessment format may influence experts' probability allocations. However, when reminded that parent state combinations presented to them were just scenarios of possible system responses, experts were able to refocus their assessment on the child state again, usually resulting in a different assessment value.

If during an elicitation we noticed the expert having difficulty allocating probabilities coherently, we tried using a budget metaphor to explain how probabilities needed to be distributed in the CPTs. Participants were told they effectively had 100 probability units for every parent instantiation. It was explained that this was like a budget that needed to be completely allocated into all available child states, with the largest number of units going to the best (most likely) child state choice for that parent instantiation. This appeared to clarify for the expert the problems that ensue under/over-specification, if the probability budget is not balanced appropriately. This usually happened during elicitation of the larger CPTs.

Subsequent to these discussions, we observed two things; 1) probability assessments were completed faster and with reduced under/over-specification error, and 2) experts became more inclined to use matrix formats. When matrix formats were adopted in this way, the expert's mode of assessment changed from one of sequential consideration of individual parent-child instantiations to a system where they considered *sets* of conditioning parent states to contextualise and iteratively re-calibrate their child node assessments on the fly. The experts appeared to first roughly rank instantiations against the available child states then allocate or calibrate individual probability allocations accordingly. In this sense the experts appeared to be mentally creating their own CPT partitioning systems to reduce the cognitive burden of large elicitation tasks.

This change of approach resulted in substantially fewer instances of what we suspect to be a positivity bias (described in following sections). These reductions were observed in both verbal and numeric response types. It is interesting to note that although one expert initially continued to use verbal responses when switching from a text-scale to a matrix format, once s/he started ranking responses as relative probabilities across the child node, numeric responses were preferred for the remainder of the interview. These observations indicate that provision of greater context may improve probability estimation. Development of interactive online tools or better utilisation of the BN GUI itself may help participants actively reorganise/rank CPTs and may be a good first step towards testing these observations more closely.

5.2 Quantifier effect

Verbal and numeric expressions of quantifiers (*few, not all, some*) and probabilities contain rhetorical and perspectival information (Moxey & Sanford 2000). Consequently, subtle but powerful information can be communicated and so can influence the inferences and responses of readers.

Moxey & Sanford go on to propose that negative quantifiers like *not all* put a different perspective on the in-

terpretation of events, which can affect the value judgment placed on the outcome.

They give the following example:

"**(10)** There is a small probability of death, which is a good*/bad thing.

(10') It is improbable that anyone will die, which is a good/bad* thing.

(11) There is a small risk of death, which is a good*/bad thing.

(11') There is an insignificant risk of death, which is a good/bad* thing."

In this example an asterisk denotes an unacceptable response. Moxey & Sanford (2000) suggest that negative quantifiers invite the reader or listener to presuppose that things are more probable or risky than they actually are. This pattern is consistent regardless of how much confidence is being expressed (e.g. *not quite certain* vs. *small probability*; Moxey & Sanford (2000)).

If the phrasing of conditioning statements can affect the perspectives and inferences of the participants, then the reasoning processes they use to generate probability assessments are also likely to be affected. This may have implications for BN knowledge engineers, because the example above is an inverse representation of the kind of conditioning statements used in probability elicitation for BNs. Rephrased as a BN elicitation query of the type used in recent research, the statement could read something like;

> "If blood alcohol level is low and the speed of the car is low, the probability of death is ____."

with participants required to complete the statement with the most accurate of the probability expressions offered. It is difficult to differentiate instances of the quantifier effect from positivity bias, which is discussed in the following section. Examples of possible instances of the quantifier effect are described in the following section on positivity bias.

5.3 Positivity bias

Teigen & Brun (2003) have shown that participants choose verbal probability phrases to correspond with the linguistic rather than the numeric content of presented information. Their experiments show that sentences containing positive quantifiers – phrases with positive directionality – will tend to receive positively framed responses, indicating that probability estimates are influenced by the way the conditioning information is presented.

Participants choose verbal phrases as a function of their frame; if they want to affirm that a particular outcome could in fact occur then they will use a term with positive directionality (e.g. 'possible', at the upper end of the verbal scale) but if the purpose is to draw attention to an events non-occurrence then a negative phrase (e.g. 'improbable', at the lower end of the scale) will be chosen (Teigen & Brun 2003). This may be because the phrases used in the text-scale worksheets and the VE software both request participants to determine "how likely" a certain response is given certain conditions. The word 'likely' creates a positive frame for the parent-child state combination requiring assessment. Positive frames may encourage positivity bias; a general readiness of participants to prefer positive over negative descriptive terms, as if positivity is the rule and negativity must be treated as an exception (Teigen & Brun 1995).

It may be possible to reduce positive framing by omitting the word 'likely' and presenting the parent-child state combination as a factual statement against which the expert applies a probability;

> "When [parent node 1] is in [state 1] and [parent node 2] is in [state 1], [child node] is [state 3]. What is the chance that this is true?"

This will be tested in future case studies. To our knowledge the presence of positivity bias in probability elicitation for BNs has not been tested directly, and may not be adequately controlled for in extant BN elicitation techniques or formats. However our observations may provide some evidence that these biases can be reduced.

For example, matrix formats (Figures 5 and 6) present experts with the entire set of parent-child state combinations all at once. So instead of considering each parent-child state combination in isolation, experts can choose to view sets of conditioning (parent) states, including the full range of possible responses (child) states across which the entirety of the 'probability budget' must be allocated. This has the advantage of making the assessment context explicit. Matrix formats may therefore provide a mechanism for participants to frame assessment requests more broadly.

5.4 Overconfidence/uncertainty avoidance

Although the apparent positivity bias was fairly easily observed, we believe a different bias was also observed during elicitations. Some experts expressed aversion to the absoluteness of the words 'certain' and 'impossible' because, in the words of one expert "nothing is certain in ecology". However, when allowed to use numeric probabilities, the same expert still responded

with 1 (certain) and 0 (impossible) values. Expressions of absolute certainty were also common in verbal responses. The result suggests that these experts may also be displaying overconfidence. There were many probabilities at the very high or very low end of the spectrum (near 1 or 0), and instances of complete disagreement were observed; where one expert assessed a parent-child state combination as 'certain' and a different expert assessed the same combination as 'impossible' (divergences between experts' assessments are discussed in more detail in the following section). This observation is similar to that described by Keren & Teigen (2001) as 'the principle of definitive predictions', or 'uncertainty avoidance', where only extreme probabilities are used in responses because participants wish to appear quite clear about what will happen next.

5.5 Aggregating expert results

Subsequent to the elicitation processes described in this paper, we aggregated (averaged) subjective probabilities and evaluated the responses in two ways. Divergences between experts' probability assessments were analysed using the relative standard deviation of the average value and the Bhattacharyya distance measure (Bhattacharyya 1943).

These techniques allowed the experts, nodes and node elements for which disagreement occurred most strongly to be identified. This is necessary so that conflicting assessments can be investigated collaboratively with participants to resolve whether the causes are clerical errors or mismatching assumptions about context – in which case the distances could be expected to diminish, or if the cause of the differences is due to contrasting conceptual models among experts – in which case structural modificatifation may be required and parallel models developed.

Further, the technique showed that although each expert provided different distributions, differences across experts occurred in equal measure. A demonstrated lack of systematic bias among experts indicated that averaging was an appropriate aggregation technique. Details of this research are reported in Thomas (2008).

6 CONCLUDING REMARKS

There is currently little guidance about how to choose between subjective elicitation methods. Preferences and responses of ecological managers and scientists were informally field-trialled using a selection of probability elicitation formats. Expert's format preferences appeared to be influenced by their familiarity with the format and the complexity of the elicitation problem. Our observations indicate that none of the trialled techniques are likely to be completely impervious to bias and overconfidence. Positively framed text-based descriptions of parent-child state combinations may have contributed to the observed bias.

Acknowledgements

This work was conducted with a scholarship from the Monash University Water Studies Centre. We gratefully acknowledge assistance from the Great Barrier Reef Marine Park Authority and the Australian Centre for Tropical Freshwater Research at James Cook University. The authors would like to thank Dr Carmel Pollino for her input and our anonymous experts for their time and good faith.

References

Aragones L.V., Lawler I.R., Foley W.J. & Marsh H. 2006. Dugong grazing and turtle cropping: grazing optimization in tropical seagrass systems? Oecologia 149(4): 635-647.

Bhattacharyya A. 1943. On a measure of divergence between two statistical populations defined by their probability distributions. Bulletin of the Calcutta Mathematical Society 35: 99-110.

Brodie J., Christie C., Devlin M., Haynes D., Morris S., Ramsay M., Waterhouse J. & Yorkston H. 2001a. Catchment management and the Great Barrier Reef. Water Science and Technology 43(9): 203-211.

Brodie J., Furnas M., Ghonim S., Haynes D., Mitchell A., Morris S., Waterhouse J., Yorkston H., Andas D., Lowe D. & Ryan M. 2001b. Great Barrier Reef Catchment Water Quality Action Plan. Townsville, Great Barrier Reef Marine Park Authority.

Brodie J., Death G., Devlin M., Furnas M. & Wright M. 2007. Spatial and temporal patterns of near-surface chlorophyll a in the Great Barrier Reef lagoon. Marine and Freshwater Research 58(4): 342-353.

Cain J. 2001. Planning improvements in natural resources management: guidelines for using Bayesian networks to support the planning and management of development programmes in the water sector and beyond. Wallingford, Oxford, Centre for Ecology & Hydrology. 132pp

Clemen R. & Winkler R.L. 1999. Combining probability distributions from experts in risk analysis. Risk Analysis 19(2): 187-203.

Costanza R., d'Arge R., de Groot R., Farber S., Grasso M., Hannon B., Limburg K., Naeem S., O'Neill R.V., Paruelo J., Raskin R.G., Sutton P. & van den Belt M. 1997. The value of the world's ecosystem services and natural capital. Nature 387: 253-260.

Crossland C.J., Done T.J. & Brunskill G.J. 1997. Potential impacts of sugarcane production on the marine environment. In: Intensive Sugarcane Production: Meeting the Challenges Beyond 2000. (Eds.) B.A. Keating & J.R. Wilson: 423-436pp.

Duarte C.M. & Chiscano C.L. 1999. Seagrass biomass and

production: a reassessment. Aquatic Botany 65(1-4): 159-174.

GESAMP 2001. Protecting the oceans from land-based activities: Land-based sources and activities affecting the quality and use of the marine, coastal and associated freshwater environment. Rep. Stud. GESAMP No. 71, IMO/FAO/UNESCO-IOC/WMO/WHO/IAEA/UN/UNEP Joint Group of Experts on the Scientific Aspects of Marine Environmental Protection and Advisory Committee on Protection of the Sea. 162pp

Hart B.T., Burgman M., Grace M., Pollino C., Thomas C. and Webb J.A. 2006. Risk-based approaches to managing contaminants in catchments. Human and Ecological Risk Assessment 12: 66-73.

Heck Jr. K.L., Hays G. & Orth R.J. 2003. Critical evaluation of the nursery role hypothesis for seagrass meadows. Marine Ecology Progress Series 253: 123-136.

Hope L.R., Nicholson A.E & Korb, K.B. 2002. Knowledge Engineering Tools for Probability Elicitation, School of Computer Science and Software Engineering, Monash University, Technical report 2002/111.

IUCN 2000. The 2000 Red List of Threatened Species. Gland, Switzerland, IUCN.

Kahneman D., Slovic P. & Tversky A., Eds. 1982. Judgement under Uncertainty, Heuristics and Biases. Cambridge, Cambridge University Press.

Keren G. & Teigen K.H. 2001. Why is p = .90 better than p = .70? Preference for definitive predictions by lay consumers of probability judgments. Psychonomic Bulletin and Review 8: 191-202.

Laskey K.B. & Mahoney S.M. 2000. Network engineering for agile belief network models. IEEE Transactions on Knowledge and Data Engineering 12(4): 487-498.

List J.A. 2001. Do explicit warnings eliminate the hypothetical bias in elicitation procedures? Evidence from field auctions for sportscards. The American Economic Review 91(5): 1498-1507.

Marsh H., Eros C., Corkeron P. & Breen B. 1999. A conservation strategy for dugongs: implications of Australian research. Marine and Freshwater Research 50: 979 – 990.

Merkhofer M.W. 1987. Quantifying judgmental uncertainty: methodology, experiences and insights. IEEE Transactions on Systems, Man and Cybernetics 17(5): 741-752.

Mumby P.J., Edwards A.J., Arias-Gonzalez J.E., Lindeman K.C., Blackwell P.G., Gall A., Gorczynska M.I., Harborne A.R., Pescod C.L., Renken H., Wabnitz C.C.C. & Llewellyn G. 2004. Mangroves enhance the biomass of coral reef fish communities in the Caribbean. Nature 427: 533-536.

Neil M., Fenton N. & Nielson L. 2000. Building large-scale Bayesian networks. The Knowledge Engineering Review 15: 257-284.

Norsys Inc. (2007). Netica, http://norsys.com.

Pollino C.A., Woodberry O., Nicholson A.E. & Korb K.B. 2005. Parameterising Bayesian networks: a case study in ecological risk assessment. (Eds.) V. Khacitvichyanukul, U. Purintrapinban and P. Uthayopas Proc. of the 2005 Int. Conf. on Simulation and Modelling (SIMMOD05), Thailand.

Rayment G. 2005. Northeast Australian experience in minimizing environmental harm from waste recycling and potential pollutants of soil and water. Communications in Soil Science and Plant Analysis 36(1-3): 121-131.

Moxey L.M. & Sanford A.J. 2000. Communicating quantities: a review of psycholinguistic evidence of how expressions determine perspectives. Applied Cognitive Psychology 14(3): 237-255.

Starck W. 2005. 'Threats' to the Great Barrier Reef. IPA Backgrounder 17(1).

Teigen K.H. & Brun W. 1995. Yes, but it is uncertain: Direction and communicative intention of verbal probabilistic terms. Acta Psychologica 88(3): 233-258.

Teigen K.H. & Brun W. 2003. Verbal probabilities: a question of frame? Journal of Behavioral Decision Making 16(1): 53-72.

Thomas C.R., Hart B.T., Nicholson A.E., Grace M. & Pollino C.A. 2005. Development Of Criteria For Simplifying Ecological Risk Models (Eds.) A. Zerger & R.M. Argent Proc. of MODSIM 2005 Int. Congress on Modelling and Simulation.

Thomas C.R. 2008. Bayesian Belief Network Development for Ecological Decision Support in Data-Sparse Domains. PhD thesis, Water Studies Centre Monash University, Melbourne. 436pp.

Ticehurst J.L., Newham L.T.H., Rissik D., Letcher R.A. & Jakeman A.J. 2007. A Bayesian network approach for assessing the sustainability of coastal lakes in New South Wales, Australia. Environmental Modelling & Software 22(8): 1129-1139.

van der Gaag L, Renooj S., Witteman C.L.M., Aleman B.M.P., & Taal B.G. 1999. How to elicit many probabilities. In UAI99 – Proc. of the Fifteenth Conf. on Uncertainty in Artificial Intelligence, Laskey and Prade, Eds., pp. 647-654.

Webb, A. 2003. Unpublished data, Water Studies Centre, Monash University.

Windschitl P. & Wells G. 1995. Measuring psychological uncertainty: verbal versus numeric methods. Journal of Experimental Psychology: Applied 2: 343-364.

Woodberry O.G., Nicholson A.E., Korb K.B. & Pollino C. 2004. Parameterising Bayesian networks. In: Lecture Notes in Computer Science. (Eds.) G.I. Webb & Y. Xinghuo. Cairns, Australia, AI2004: Advances in Artificial Intelligence: 17th Australian Joint Conf. on Artificial Intelligence: p 1101-1107.

Zimmer A.C. 1984. A model for the interpretation of verbal predictions. International Journal of Man-Machine Studies 20: 121-134.

Figure 6: Example matrix worksheet, adapted from Laskey & Mahoney (2000).

Relieving the elicitation burden of Bayesian Belief Networks

B.W. Wisse, S.P. van Gosliga, N.P. van Elst, A.I. Barros
TNO Defence, Security and Safety
P.O. Box 96864, 2509JG The Hague
The Netherlands

Abstract

In this paper we present a new method (EBBN) that aims at reducing the need to elicit formidable amounts of probabilities for Bayesian belief networks, by reducing the number of probabilities that need to be specified in the quantification phase. This method enables the derivation of a variable's conditional probability table (CPT) in the general case that the states of the variable are ordered and the states of each of its parent nodes can be ordered with respect to the influence they exercise. EBBN requires only a limited amount of probability assessments from experts to determine a variable's full CPT and uses piecewise linear interpolation. The number of probabilities to be assessed in this method is linear in the number of conditioning variables. EBBN's performance was compared with the results achieved by applying both the normal copula vine approach from Hanea & Kurowicka (2007), and by using a simple uniform distribution.

1 Introduction

In this paper we consider the case of deriving a discrete conditional probability distribution for a node of a Bayesian belief network based on expert judgement. There are many issues to consider when deriving a conditional probability distribution via expert judgement elicitation. The expert assessors will for example use simplifying heuristics when assessing probabilities to avoid too complex mental reasoning. These heuristics might lead to biased assessments. In addition experts might also be subject to various types of motivational biases. There is the problem of how to select the appropriate experts for the elicitation task and how to properly prepare them for formulating the assessments (e.g. motivating and training them). There is the choice of which method to use to elicit the probabilities: e.g. a probability-scale, probability-wheel, gamble-like or adverb-probability matching method? Renooij (2001) gives a good overview of these issues. Though acknowledging their importance, in this paper we do not consider these issues, but focus on reducing the assessment burden of large discrete conditional probability distributions.

The number of probabilities that need to be specified for a node can grow large very easily. For a node with three states that has a parent node with also three states, 6 probabilities need to be specified to determine its conditional probability table (CPT). An additional second and third parent node with three states would consequently require a table of 18 and 54 probabilities, and so on. Apart from the huge amounts of time it would take to assess all the probabilities for large CPTs, it can also be questioned to what extent assessors can be expected to coherently provide the probabilities at the level of detail required (see e.g. (Miller 1956) on the limitations of human short term memory capacity). The elicitation task thus is considered a major obstacle in the use of BBNs (Druzdzel & Van der Gaag 1995, Jensen 1995).

There are two ways in which the elicitation task for discrete BBNs can be relieved. The first is to make it easier for the assessors to provide the probabilistic assessments required. Van der Gaag, Renooij, Witteman, Aleman & Taal (1999) aim to achieve this by transcribing the conditional probabilities and using a scale containing both numerical and verbal anchors. But the effort needed to assess a full CPT using this method, though reduced, is still exponential in the number of conditioning variables. The second option for relieving the elicitation burden is to reduce the number of probabilistic assessments to be made. This can of course be achieved by reducing the number of conditions (parent nodes) or the number of states of the variables, but such reductions will often be undesirable (e.g. leading

to loss of detail needed to inform a decision).

The Noisy-OR model, originally introduced by Kim & Pearl (1983), but more extensively discussed in relation to BBNs by e.g. Heckerman & Breese (1996), reduces the number of probabilities to be specified by making additional assumptions about the underlying causal structure of the variables. For the noisy-OR model, the number of probabilities needed to determine the full CPT is linear in the number of conditioning variables, rather than exponential. Although this can mean a huge reduction in elicitation effort, the assumptions necessary are strong and all the variables in the noisy-OR model need to be binary, which strongly limits the applicability of the method.

The Noisy-MAX model (Díez 1993) can be seen as the extension of the Noisy-OR to multi-valued variables. In this model the CPT is derived from 'marginal conditional' distributions specified for each parent: for each parent the probabilities conditional on this parent node are specified and subsequently the full CPT is derived from these conditional probabilities using the max function. The influences of each of the parent nodes are treated in this model as independent. So the joint influence that the parent nodes exercise is fully determined by their marginal influence and a fixed function. Zagorecki & Druzdzel (2006) have fitted the Noisy-MAX model to suitable nodes from three belief networks for which the CPTs where already specified. The authors found the model to be able to provide a good fit to the CPT in about 50% of the cases they considered.

In this paper we develop and evaluate a methodology, EBBN, for deriving a node's CPT in the general case that the states of the node are ordered and the states of each of its parent nodes can be ordered with respect to the influence these parent nodes have on this node of interest. In this method only a (small) part of the CPT- describing the joint influence of the parents in contrast with the marginal influence elicited in the Noisy-MAX model - is elicited. The conditional probabilities that are not directly elicited are derived using an interpolation method based on the ranks of parent node states. The number of probabilities to be assessed is linear in the number of parent nodes. Since the method approximates the probabilities that are not directly assessed, it will contain inaccuracies. Like Van der Gaag et al. (1999) we therefore propose to regard and use this method as a first step in an iterative procedure of stepwise refinement of probability assessments, like described in (Coupé, Peek, Ottenkamp & Habbema 1999).

While testing this method three relevant alternatives were presented. Bonafede & Giudici (2007) have developed a method for deriving a discrete conditional probability distribution based on the marginal distributions, correlation coefficients and standardised joint moments. Yet, this method also requires all the variables to be binary, and closed-form solutions have only been derived for up to three conditioning variables (parent nodes). Secondly Hanea & Kurowicka (2007) provide a method for determining a CPT based on the copula vine approach (Bedford & Cooke 2002) that uses similar prior information: marginal distributions and adjusted (conditional) rank correlations. This method also provides a means for deriving the CPT in the general case that the variables are ordinal and the influences are monotone, although it is not clear to us if and how the prior assessments needed can be elicited accurately from experts. In Section 4 we compare the results of the method developed in this paper with the copula vine approach of Hanea & Kurowicka, for which the required prior assessments are derived from a fully specified CPT.

Very closely related to our method is the method presented by Tang & McCabe (2007). These authors also propose the use of piecewise linear interpolation to approximate not-elicited conditional probabilities. Furthermore they introduce the concepts of dominant and important factors, whilst we use positive and negative dominance and parent weights. Yet, where Tang & McCabe, like Bonafede & Giudici, restrict their method to work with binary variables only, the method we introduce in this paper works with discrete variables in general, under the above described conditions of ordinality of the variables. It should be noted that the development of our method has taken place independently of that of Tang & McCabe.

In the next section we will introduce our alternative elicitation method for BBNs, EBBN, which is aimed at reducing the elicitation burden. In Section 3 we discuss when we can regard an approximation of a CPT to be 'good', providing the means to assess the performance of the proposed method and compare it with the copula vine approach (Section 4). In the final section we present our conclusions and suggestions for future work.

2 The EBBN Method

We regard the problem of expert assessment of the probability distribution of a discrete variable X_c (a node in a BBN), conditional on a set of two or more discrete variables, which we will denote with $pa(X_c)$ (the set of parent nodes). We require (1) the values of X_c to be ordered, and (2) that the values of each of the elements of $pa(X_c)$ can be ordered such that each of these variables have either a positive or a negative

influence on X_c. By stating that $X_k \in pa(X_c)$ has a positive influence on X_c, denoted by $S^+(X_k, X_c)$, we mean that observing a higher value for X_k does not decrease the likelihood of higher values of X_c, regardless of the values of the other variables $pa(X_c) \setminus X_k$. We take assignment $a = \{x_j, \ldots, x_u\}$ to be an instantiation of the set of $pa(X_c) = \{X_j, \ldots, X_u\}$. Formally we define $X_k \in pa(X_c)$ having a positive influence on X_c, $S^+(X_k, X_c)$, as (Wellman 1990): for all values x_c of X_c, for all pairs of distinct values $x_{k,n} > x_{k,o}$ of X_k, and for all possible assignments $a_{\neg k}$ for the set of $pa(X_c) \setminus X_k$,

$$P(X_c > x_c \mid x_{k,n}, a_{\neg k}) \geq P(X_c > x_c \mid x_{k,o}, a_{\neg k}).$$

The definition of a negative influence, $S^-(X_k, X_c)$, is completely analogous and would involve only reversing the above inequality.

We define a conditioning variable $X_k \in pa(X_c)$ to be *positive dominant*, if the following two (sets of) assignments of $pa(X_c)$ lead to the same probabilities: (I) all assignments of $pa(X_c)$ in which X_k is in its most favourable state for high values of X_c and (II) the assignment in which each $X_l \in pa(X_c)$ is in its most favourable state for higher values of X_c (i.e. all $X_p \in pa(X_c)$ with $S^+(X_p, X_c)$ are at their highest value, and all $X_n \in pa(X_c)$ with $S^-(X_n, X_c)$ are at their lowest value).

So if a positive dominant parent is in its most favourable state for high values of X_c, then, regardless of the states of the other parents, X_c will have the same probabilities as when conditional on the assignment in which all parent nodes are in their most favourable state. *Negative dominant* variables are defined analogously.

In the remainder of this section we will first discuss the assessments needed from the expert for the derivation of the CPT of X_c. We will then show how to obtain the CPT from these assessments and end the section with an illustrative example of the method, taken from the Hailfinder network (Abramson, Brown, Edwards, Murphy & Winkler 1996).

2.1 Required assessments

It is assumed that the assessor has confirmed that the values of variable X_c are ordered and that the assessor can order the values of each of the variables $X_k \in pa(X_c)$ such that (s)he judges either $S^+(X_k, X_c)$ or $S^-(X_k, X_c)$ to hold. Then the following assessments are required to determine the CPT for variable X_c with conditioning variables $pa(X_c)$[1]:

1. (ordering). For each of the conditioning variables $X_k \in pa(X_c)$: order the values of X_k such that X_k has either a positive or a negative influence on X_c. Fix and record this ordering of the values and the nature of the influence.

2. (typical probabilities). For each of the values x_c of X_c:

   (a) determine the assignment $pa(X_c) = a_{x_c}$ such that the probability $P(X_c = x_c \mid a_{x_c})$ is as large as possible.

   (b) assess the probabilities $P(X_c \mid a_{x_c})$.

   Due to dominance of one of the conditioning variables $a_{x_{c,min}}$ ($a_{x_{c,max}}$) need not be unique, where $x_{c,min}$ ($x_{c,max}$) is the lowest (highest) value of X_c. Therefore $a_{x_{c,min}}$ ($a_{x_{c,max}}$) is by default set to be the assignment in which all the conditioning variables are in their most favourable state for low (high) values of X_c, referred to as a_{neg} (a_{pos}).

3. (weights). For each of the conditioning variables $X_k \in pa(X_c)$, assess $P(X_c = x_{c,max} \mid a_{neg,k+})$ and $P(X_c = x_{c,min} \mid a_{neg,k+})$, where $x_{c,max}$ and $x_{c,min}$ are resp. the maximum and minimum value of X_c, and $a_{neg,k+}$ is the assignment of $pa(X_c)$ in which X_k is in its most favourable state for high values of X_c, and all $X_l \in pa(X_c) \setminus X_k$ are in their least favourable state for higher values of X_c.

4. (dominance). For each of the conditioning variables $X_k \in pa(X_c)$, determine whether X_k has either no, a positive or a negative dominance over X_c.

2.2 Deriving the CPT

The derivation of the CPT of X_c is done in a two-step procedure, using the assessments from Section 2.1. In the first step we will express the probabilities $P(X_c)$ as a function of an influence factor i. In the second step individual and joint influence factors are determined for all assignments of $pa(X_c)$, which are then used to derive the probabilities $P(X_c)$ from the functions of step 1.

The influence factor i is an expression of the positiveness (or negativeness) of the joint influence of the parent variables $pa(X_c)$ on X_c. It is a function of values of the parent variables, with $0 \leq i(a) \leq 1$. We set $i(a_{neg}) = 0$, where $pa(X_c) = a_{neg}$ is the assignment in which all the conditioning variables are in their most favourable state for low values of X_c (see item 2, Section 2.1). And, at the other extreme, $i(a_{pos})$ is set to 1. For all other assignments $i \in (0, 1)$. If assignment a_2 has a strictly more positive influence on X_c than a_1

[1] As mentioned in Section 1 we will not discuss here how these assessments can be best elicited from the assessor

- i.e. $P(X_c > x_c \mid a_2) > P(X_c > x_c \mid a_1)$ for all x_c - then the influence factor corresponding to a_2 should be bigger than the influence factor corresponding to a_1.

We make use of two separate influence factors: the *individual influence factor* i_k for each conditioning variable $X_k \in pa(X_c)$ and the *joint influence factor* i_{joint}. As will become more clear later on, i_k will contain information about the influences exercised by each of the parent variables individually, i_{joint} about the 'general tendency' of all of the parent influences together.

We determine the individual influence factor i_k for $X_k \in pa(X_c)$ as follows:

$$i_k(x_k) := \begin{cases} \dfrac{\text{rank}(x_k) - 1}{\text{rank}(x_{k,max}) - 1} & \text{if } S^+(X_k, X_c) \\ \dfrac{\text{rank}(x_{k,max}) - \text{rank}(x_k)}{\text{rank}(x_{k,max}) - 1} & \text{if } S^-(X_k, X_c) \end{cases} \tag{1}$$

where the rank of the smallest value is set to be 1 and $x_{k,max}$ is the highest value of X_k. So if $X_k \in \{low, medium, high\}$ has a positive influence on X_c, we find that $i_k(low) = 0$, $i_k(medium) = 0.5$ and $i_k(high) = 1$.

The joint influence factor i_{joint} for assignment $pa(X_c) = a$ is derived as:

$$i_{joint}(a) := \frac{\sum_{\{k:X_k \in pa(X_c)\}} i_k(x_k) \cdot (\text{rank}(x_k) - 1)}{\sum_{\{k:X_k \in pa(X_c)\}} (\text{rank}(x_{k,max}) - 1)} \tag{2}$$

Verify that indeed $i_{joint}(a_{neg}) = 0$ and $i_{joint}(a_{pos}) = 1$. Also note that the individual influence factor of X_k, i_k, is equal to the joint influence factor i_{joint} if $pa(X_c) = \{X_k\}$, i.e. if the set of parents of X_c merely consists of X_k.

Step 1. Estimating $P(X_c)$ as a function of joint influence factor i_{joint}

In this step $P(X_c = x_c)$ is estimated as a function of joint influence factor i_{joint}, for each value x_c of X_c. For this we use the orderings determined at item 1 in Section 2.1, and the assignments a_{x_c} and probabilities $P(X_c = x_c \mid a_{x_c})$ assessed at 2. We construct the piecewise linear functions $f_{x_c} : [0,1] \rightarrow [0,1]$ through the points $(i_{joint}(a_{x_c}), P(X_c = x_c \mid a_{x_c}))$. It can be easily verified that using these linear interpolations ensures that $\sum_{x_c} f_{x_c}(i) = 1$, i.e. the sum of the probabilities of occurrence of the different values of X_c equals unity for all $i \in [0,1]$. Coherency requires that if $x_{c,n} > x_{c,m}$, also $i_{joint}(a_{x_{c,n}}) > i_{joint}(a_{x_{c,m}})$. In Figure 1 an example is given for how this estimation of $P(X_c)$ as a function of i_{joint} might look like. In this example $X_c \in \{low, medium, high\}$, and the points $(i_{joint}(a_{x_c}), P(X_c = x_c \mid a_{x_c}))$ are assessed as in Table 1.

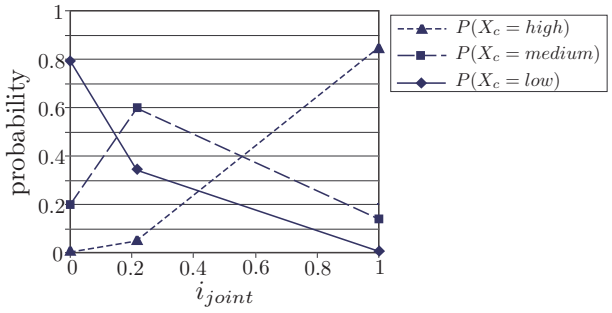


Figure 1: Piecewise linear functions through the points $(i_{joint}(a_{x_c}), P(X_c \mid a_{x_c}))$ from Table 1.

Table 1: Example assessments of $(i_{joint}(a_{x_c}), P(X_c \mid a_{x_c}))$ for $X_c \in \{low, medium, high\}$

x_c	$i_{joint}(a_{x_c})$	$P(X_c \mid a_{x_c})$
low	0	$P(X_c = low \mid a_{low}) = 0.79$ $P(X_c = medium \mid a_{low}) = 0.20$ $P(X_c = high \mid a_{low}) = 0.01$
$medium$	0.22	$P(X_c = low \mid a_{medium}) = 0.35$ $P(X_c = medium \mid a_{medium}) = 0.60$ $P(X_c = high \mid a_{medium}) = 0.05$
$high$	1	$P(X_c = low \mid a_{high}) = 0.01$ $P(X_c = medium \mid a_{high}) = 0.14$ $P(X_c = high \mid a_{high}) = 0.85$

Note that $pa(X_c) = a_{low}$ corresponds to the assignment $pa(X_c) = a_{neg}$ and a_{high} to a_{pos}. Hence $i_{joint}(a_{low}) = 0$ and $i_{joint}(a_{high}) = 1$.

Step 2. Deriving the conditional probabilities

In Step 1 we obtained $P(X_c)$ for all possible values of i_{joint} via linear interpolation, and equation (2) provides us with an expression for i_{joint} for all assignments $pa(X_c) = a$. We can now determine $P(X_c \mid a)$ via $P(X_c \mid i_{joint}(a))$ from the functions f_{x_c} of Step 1. Yet this mapping from assignments a for the conditioning variables $pa(X_c)$ to an expression i_{joint} is not unique. Suppose $pa(X_c) = \{X_j, X_k, X_l\}$, X_j and X_l both exercise the same type of influence (positive or negative), and $X_j, X_k, X_l \in \{low, medium, high\}$, then $i_{joint}(\{medium, medium, medium\}) = i_{joint}(\{low, medium, high\}) = 0.5$. As pointed out earlier, i_{joint} is an expression for the 'general tendency' of the influence of the conditioning variables. It does not take into account the (dis)agreement of the influences of each of the conditioning variables individually.

To account for both the 'general tendency' and the individual influences of the conditioning variables, we

calculate for each conditioning variable $X_k \in pa(X_c)$ the average of the probabilities $P_k(X_c \mid a)$ over the interval $(min(i_k(x_k), i_{joint}(a)),\ max(i_k(x_k), i_{joint}(a)))$. An example of this average, denoted with $\overline{P_k(X_c \mid a)}$, is illustrated in Figure 2.

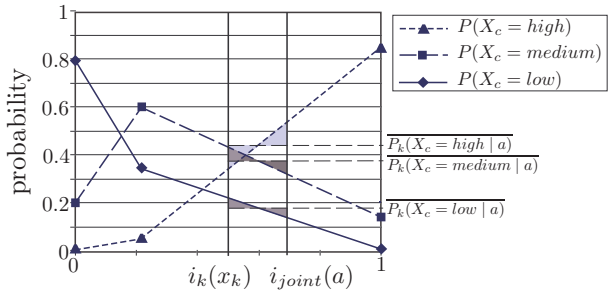


Figure 2: Example of the average probabilities $\overline{P_k(X_c)}$, when $i_k(x_k) < i_{joint}(a)$

We derive the desired probabilities $P(X_c \mid a)$ as the average over the distributions $\overline{P_k(X_c \mid a)}$. Or actually the *weighted* average

$$P(X_c \mid a) = \sum_{k:X_k \mid pa(X_c)} w_k \cdot \overline{P_k(X_c \mid a)}, \qquad (3)$$

since one parent could have a stronger influence on X_c than another. For the same relative change in states, i.e. changes in states resulting in the same absolute change in each of the individual influence factors, the probabilities for X_c might change more for one parent variable than for another. Therefore we calculate the weight w_k for each parent $X_k \in pa(X_c)$, in the following way:

$$w_k = \frac{1}{2}\frac{\delta_k^+}{\sum_{l:X_l \in pa(X_c)} \delta_l^+} + \frac{1}{2}\frac{\delta_k^-}{\sum_{l:X_l \in pa(X_c)} \delta_l^-} \qquad (4)$$

with,

$$\delta_k^+ = P(X_c = x_{c,max} \mid a_{neg,k+}) - P(X_c = x_{c,max} \mid a_{neg})$$
$$\delta_k^- = P(X_c = x_{c,min} \mid a_{neg}) - P(X_c = x_{c,min} \mid a_{neg,k+}).$$

For the derivation of the weights we have taken the situation in which each parent is in its least favourable state for high values of X_c, a_{neg}, as the base. We use the probabilities $P(X_c = x_{c,max} \mid a_{neg,k+})$ and $P(X_c = x_{c,min} \mid a_{neg,k+})$ assessed at item 3 in Section 2.1. Each δ_k^+ and δ_k^- now expresses the changes in the probabilities of resp. the highest and lowest state of X_c, if the one parent X_k is set to its most favourable state for high values of X_c whilst leaving the other parents in their least favourable states ($a_{neg,k+}$). We obtain the weights from these δ's via the normalisations (4).

To a large extent the choice of the base assignment a_{neg} and the probabilities $P(X_c = x_{c,max} \mid a_{neg,k+})$ and $P(X_c = x_{c,min} \mid a_{neg,k+})$ to derive the weights is arbitrary. Even though, we feel the choice for these assignments is one of the most natural choices that can be made. And, more importantly, we feel these assignments are relatively easy for assessors to consider and assess. It is of course possible to use more assessments to determine the weights more accurately. However, we feel that the possible added value does not weigh against the burden of the extra elicitation effort needed.

We derive the desired probabilities $P(X_c \mid pa(X_c) = a)$ by rewriting (3) using (1), (2) and (4), as

$$P(X_c \mid pa(X_c) = a) = \sum_{k:X_k \mid pa(X_c)} w_k \cdot \frac{\int_{i_{min,k}}^{i_{max,k}} \mathbf{f}(i) \cdot di}{i_{max,k} - i_{min,k}} \qquad (5)$$

where $i_{min,k} = min(i_k(x_k), i_{joint}(a))$, $i_{max,k} = max(i_k(x_k), i_{joint}(a))$ and $\mathbf{f}(i) = \Big(f_{x_{c,min}}(i), \ldots, f_{x_{c,max}(i)}\Big)$.

Finally, we deal with negative and positive dominance of one of the parent variables in the following straightforward way: for all the assignments a_d in which a negative (positive) dominant parent is in its least (most) favourable state for high values of X_c, we set $P(X_c \mid a_d)$ to be equal to $P(X_c \mid a_{neg})$ ($P(X_c \mid a_{pos})$). We will now demonstrate the method by means of an illustrative example.

2.3 Illustrative example from the Hailfinder network

The example given in this section is based on the CompPlFcst variable from the Hailfinder network (Abramson et al. 1996). The variable and its parent nodes, AreaMeso_ALS, CldShadeOth, CldShadeConv and Boundaries, are depicted in Figure 3. For each of the variables also the states (discrete values) are given, ordered and with the highest state on top.

For the variable CompPlFcst we have the fully subjectively specified CPT, consisting of $4 \cdot 3^3 \cdot 3 = 324$ probabilities. In this example we derive the required assessments for EBBN, as specified in Section 2.1, from this CPT, but treat them as if they were directly elicited:

1. (ordering). The ordering of the states of the variables is given in Figure 3, where the highest states are on top. For the conditioning variables we find the following influences:
   S^-(AreaMeso_ALS,CompPlFcst); S^+(CldShadeOth,CompPlFcst);
   S^-(CldShadeConv,CompPlFcst); S^+(Boundaries,CompPlFcst).

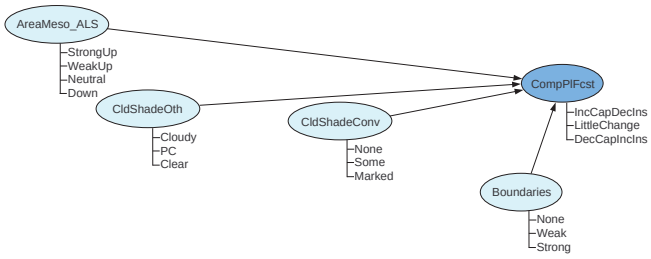


Figure 3: The variable CompPlFcst and its parent nodes from the Hailfinder network.

2. (typical probabilities). We find the assignments: $a_{DecCapIncIns}$ ={*StrongUp*, *Clear*, *None*, *Strong*}, $a_{LittleChange}$ ={*StrongUp*, *PC*, *Some*, *Strong*} and $a_{IncCapDecIns}$ ={*Down*, *Cloudy*, *Marked*, *None*}.
   The corresponding conditional probabilities are given in Table 1 and depicted as a function of influence factor i in Figure 1, where $a_{low} = a_{DecCapIncIns}$, $a_{medium} = a_{LittleChange}$ and $a_{high} = a_{IncCapDecIns}$.

3. (weights). As assessments of the remaining probabilities needed to derive the parent weights we find:

$a_{neg,k+}$	$P(X_c = x_{c,min} \mid a_{neg,k+})$	$P(X_c = x_{c,max} \mid a_{neg,k+})$
$a_{neg,\text{AreaMeso\_ALS}+}$	0.20	0.45
$a_{neg,\text{CldShadeOth}+}$	0.40	0.30
$a_{neg,\text{CldShadeConv}+}$	0.52	0.13
$a_{neg,\text{Boundaries}+}$	0.65	0.05

4. (dominance). No (positive or negative) dominant parents.

Now we have all the information (containing only 17 probability assessments!) we need to derive all the 324 probabilities of the CPT of CompPlFcst.

By means of an example we calculate the probabilities $P(\text{CompPlFcst} \mid pa(\text{CompPlFcst}) = a_{expl})$, where a_{expl} ={AreaMeso_ALS=*Down*, CldShadeOth=*PC*, CldShadeConv=*None*, Boundaries=*Strong*}. For these parent node states we find the individual influence factors: $i_{\text{AreaMeso\_ALS}}(Down) = 1$, $i_{\text{CldShadeOth}}(PC) = \frac{1}{2}$, $i_{\text{CldShadeConv}}(None) = 0$ and $i_{\text{Boundaries}}(Weak) = 0$, and a joint influence factor $i_{joint}(a_{expl}) = \frac{4}{9}$. So we see in this case that the individual influence factors of the parents give a diverse picture, two are very negative (0), one is between negative and positive ($\frac{1}{2}$) and one is very positive (1). This is reflected by the joint influence factor, which has a very average value (0.44), expressing no general tendency of the parent influences towards either positive or negative influence.

Based on the assessments and (4), we find the weights: $w_{\text{AreaMeso\_ALS}} = 0.459$, $w_{\text{CldShadeOth}} = 0.303$, $w_{\text{CldShadeConv}} = 0.165$ and $w_{\text{Boundaries}} = 0.073$. We can now use (5) to derive the desired probabilities and find $P(\text{CompPlFcst} \mid pa(\text{CompPlFcst}) = a_{expl}) = \{0.17, 0.32, 0.51\}$. We can derive the full CPT of X_c (consisting of 324 probabilities) in the same way, requiring in this case only 17 probabilities to be assessed. When we look up the probabilities in de original CPT, we find $P(\text{CompPlFcst} \mid pa(\text{CompPlFcst}) = a_{expl}) = \{0.20, 0.32, 0.48\}$. The probabilities estimated with the methodology are in this case 'not far off'. Yet, before we can assess how well our method approximates the directly assessed probabilities, we first need to discuss how we can measure the quality of the approximation.

3 Approximation of a CPT for a BBN, when is it 'good'?

Assuming you have knowledge of the 'true' probabilities of a certain CPT, how can you assess the quality of an approximation to that CPT? A measure to assess the similarity between two (discrete conditional) probability distributions, with possibly different support, is the Jensen-Shannon divergence (Lin 1991). Based on the Kullback-Leibler divergence, this measure does not take into account the context of the CPT, the belief network. Both Henrion (1989) and Chan & Darwiche (2002) show that inference in a belief network is most sensitive to assessment errors in probabilities that are close to zero or one.

Druzdzel & Van der Gaag (2000) state that, since inaccuracies will influence the output of the belief network, a natural question to ask is how accurate the approximation should be to arrive at satisfacory behaviour of the network. In other words: if the network is constructed to perform specific queries, does the use of approximations still lead to acceptable outcomes on these queries?

Chan & Darwiche (2002) identify three main approaches in the literature to measure the impact of a change in probability in a CPT: measuring the absolute change in the probability of a query, the relative change in the probability of a query or the relative change in the odds of the query, finding the first to be the most prevalent in the literature.

Zagorecki & Druzdzel (2006) give two measures to express the (dis-)similarity of two CPTs for the same conditional distribution: the Euclidian distance and the Kullback-Leibler divergence between the two CPTs. Time and space unfortunately have prohibited us to implement these measures in the current investi-

gation. We have used the following measures to assess the performance of the EBBN in the next section:

m1. Average absolute error in probability.

m2. Average Jensen-Shannon divergence: a measure of the similarity between the 'true' CPT and the approximation to it.

m3. Maximum Jensen-Shannon divergence.

m4. Number of unmatched certainties and impossibilities: the number of times the 'true' and the approximating CPT disagree on probabilities of 0 and 1. As noted above, queries can be very sensitive to extreme probabilities.

m5. % agreement in likelihood ranking: the percentage of scenarios in which the likelihood ranking of the values of the variable is the same for both the 'true' CPT as the approximating CPT. As scenarios all logically possible combinations of values of the neigbouring (i.e. predecessor and descendent) nodes are taken.

m6. % agreement on most likely state: the percentage of scenarios in which the most likely state for the variable is the same for both the 'true' CPT as the approximating CPT. As scenarios all logically possible combinations of values of the neigbouring (i.e. predecessor and descendent) nodes are taken.

4 Performance of EBBN

We have investigated the performance of the methodology by applying it to a well-known belief network from the literature that contained suitable large subjectively assessed CPTs, and comparing its performance with the copula vine approach from Hanea & Kurowicka (2007). We found the Hailfinder network (Abramson et al. 1996) to contain such CPTs.

4.1 Methodology

We have searched for belief networks that contained nodes that satisfy the following requirements:

- the CPT of the node was subjectively assessed,
- the CPT of the node has to be reasonably challenging in size for elicitation from an expert. For this we decided the node needed to have two or more parents, and
- the states of the node are ordered.

We found these networks are difficult to come by. This is not surprising of course, since these networks would require a huge elicitation effort. Practitioners would usually try to avoid having to specify these large CPTs because the elicitation process would be too time consuming, the very problem we are aiming to deal with in this article. In the BBN repository of the University of Pittsburgh[2] we found the Hailfinder network, which does contain 7 nodes that satisfy our requirements.

For the Hailfinder network we created three alternative versions. In each of these alternative versions we replaced the CPTs of the 7 nodes satisfying the above requirements (and kept the remaining CPTs as they were). In the first alternative implementation these CPTs were replaced with the approximations resulting from the method introduced in this paper. We treat the CPTs from the literature as the 'true' CPTs. We assume that the probabilities needed for our methodology would have been assessed as they are in these CPTs and treat the difference between approximations of the method and the corresponding CPTs as inaccuracies of the approximation. So we have *not* tried to find parameters for EBBN that minimise the distance of the resulting CPT to the original, but have derived the parameters needed from the original CPT.

The second alternative implementation has the selected 7 CPTs derived according to the copula vine approach (Hanea, Kurowicka & Cooke 2006). In this approach a normal copula vine is constructed based on the marginal distributions of each variable and its conditioning variables (or actually continuous versions of these discrete marginals) and (conditional) rank correlation coefficients of the variable with each of its conditioning variables. This normal copula vine specifies a joint distribution of the variable and its conditioning variables. Hanea & Kurowicka (2007) describe how the (conditional) rank correlation coefficients can be derived from a CPT. If one was to use the copula vine approach in practice, the marginal distribution of the variable under consideration and the (conditional) rank correlations with each of the conditioning variables would have to be subjectively assessed, which is not a trivial task. Since we are using the copula vine approach as a benchmark here, as a different means of approximating the 'true' CPT, we simply derived this marginal and the correlations from the 'true' CPT. The used marginal and correlations thus represent the best values that could have been obtained in an elicitation process. After construction we took a large sample (we used a sample size of 80,000) from the normal

[2]The belief network models can be found at the network repository of the Decision Systems Laboratory of the Univesity of Pittsburgh (http://genie.sis.pitt.edu/networks.html)

copula vine and estimated the desired copula vine version of the CPT from the frequencies in this sample. We checked that the marginal of the variable under consideration and the marginals of its parents were still as specified for the copula.

Finally we constructed a third alternative implementation of the Hailfinder network in which all altered CPTs consist of uniform distributions for all assignments of conditioning variables, to serve as a second benchmark. We have assessed the performance of our interpolation method and both benchmarks using the measures specified in Section 3. We found the variable InsInMt to be a positive dominant parent of CldShadeConv, and treated it as such in all three alternative implementations.

4.2 Results

The results of the comparison of the 'true' versions of the selected 7 CPTs of the Hailfinder network with each of the three alternative derivations of these CPTs are given in Table 2 [3]. The table displays how the EBBN, copula vine and the uniform versions of the CPTs score on 9 performance measures. The first four measures are the measures m1.-m4. from Section 3 for the direct comparison between the 'true' and the approximating versions of the CPTs. The measures in the last five columns, m1.-m3., m5. and m6. from Section 3, consider posterior probabilities for each of the 7 selected nodes under all possible scenarios for neighbouring nodes, i.e. all logically possible combinations of states of neighbouring nodes (both parent and child nodes).

For the first seven measures in the table we have that the smaller the measure, the better the performance of the approximating CPT on that measure. For the last two columns to opposite holds: the higher the percentage, the better the performance. If a number is underlined in Table 2, this means that the corresponding approximating method (EBBN, copula vine or uniform) has the best performance for that measurement on that variable.

If we look at the underlined values in Table 2, it seems that EBBN and the copula vine versions are of comparable performance on all performance measures apart from 'unmatched 0/1', on which EBBN performs best on all CPTs. It is comforting to see that both EBBN and the copula vine approach clearly perform better than when the CPT is populated with merely uniform distributions. Further investigation reveals that the EBBN method scores relatively well on the so called 'collector' variables CombMoisture, CombVerMo and CombClouds. These are nodes in the Hailfinder network that "summarize information from different sources about moisture, vertical motion and clouds, respectively" (Abramson et al. 1996, p.69). The EBBN method seems a relatively good means to combine similar information from different sources, at least for the Hailfinder network.

[3] The EBBN and copula vine versions of the Hailfinder network (.xdsl format) can be obtained from the authors.

5 Conclusions and discussion

In this paper we have developed a method for deriving large conditional probability tables based on expert judgement, that can hugely reduce the number of assessments needed from the experts. The quantitative assessments needed from the experts are relatively easy to understand: the experts still need to assess only probabilities. We believe that the experts will also be capable of providing the qualitative judgements described in Section 2.1 at items 1, 2(a) and 4.

In order to evaluate the performance of EBBN we applied it to a well-known belief network from the literature, the Hailfinder network. EBBN's performance was compared with the results achieved by applying both the normal copula vine approach from Hanea & Kurowicka (2007), and by using a simple uniform distribution. The results show that EBBN's performance is comparable to the the performance of the normal copula vine approach, and distinctly better than that of the uniform distributions. We believe that the EBBN method can be a valuable tool for subjectively specifying large CPTs.

In the development of the method, the application to a real-life example (the Hailfinder network) has proven very valuable. We would like to test the method on more examples. But, as noted before, because of cost and effort required to elicit large CPTs, these examples are difficult to find. Any help with finding more examples would be greatly appreciated.

It should be noted that the EBBN method does not always lead to a large reduction in the number of probabilities that need to be assessed. In fact, the method could even require more probabilities to be assessed than there are in the CPT. Roughly this occurs when the number of states of the variable for which the CPT is to be derived is greater than the number of conditions (i.e. the number of different assignments of the conditioning variables).

Finally we would like to remark that the interpolation used, in its current form, does not take into account synergetic effects that may exist between conditioning variables.

Acknowledgements

We are grateful to the Institute of Applied Mathematics of the TU Delft for allowing us to use their Unicorn and Uninet software to derive the normal copula vine versions of the 7 CPTs of the Hailfinder network.

References

Abramson, B., Brown, J., Edwards, W., Murphy, A. & Winkler, R. L. (1996), 'Hailfinder: a Bayesian system for forecasting severe weather', *International Journal of Forecasting* **12**, 57–71.

Bedford, T. J. & Cooke, R. M. (2002), 'Vines - a new graphical model for dependent random variables', *Annals of Statistics* **10**(4), 1031–1068.

Bonafede, C. & Giudici, P. (2007), 'Bayesian networks for enterprise risk assessment', *Physica A* **382**, 22–28.

Chan, H. & Darwiche, A. (2002), 'When do numbers really matter?', *Journal of Artificial Intelligence Research* (17), 265–287.

Coupé, V. M., Peek, N., Ottenkamp, J. & Habbema, J. D. F. (1999), 'Using sensitivity analysis for efficient quantification of a belief network', *Artificial Intelligence in Medicine* **17**, 223–247.

Díez, F. (1993), Parameter adjustment in Bayesian networks. the generalized noisy or-gate, *in* 'Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI93)', Morgan Kaufmann, pp. 99–105.

Druzdzel, M. J. & Van der Gaag, L. C. (1995), Elicitation of probabilities for belief networks: combining qualitative and quantitative information, *in* 'Proceedings of the eleventh conference on uncertainty in artificial intelligence', pp. 141–148.

Druzdzel, M. J. & Van der Gaag, L. C. (2000), 'Building probabilistic networks: where do the numbers come from?', *IEEE Transactions on Knowledge and Data Engineering* **12**(4), 481–486.

Hanea, A. & Kurowicka, D. (2007), Mixed nonparametric continuous and discrete Bayesian belief nets, *in* 'Proceedings of the Fifth International Mathematical Methods in Reliability (MMR) Conference', Glasgow.

Hanea, A., Kurowicka, D. & Cooke, R. M. (2006), 'Hybrid method for quantifying and analyzing bayesian belief nets', *Quality and Reliability Engineering International* **22**(6), 613–729.

Heckerman, D. & Breese, J. (1996), 'Causal independence for probabilistic assessment and inference using Bayesian networks', *IEEE Transactions on Systems, Man and Cybernetics* **26**, 826–831.

Henrion, M. (1989), 'Some practical issues in constructing belief networks', *Uncertainty in Artificial Intelligence* **3**, 161–173.

Jensen, A. L. (1995), Quantication experience of a dss for mildew management in winter wheat, *in* M. J. Druzdzel, L. C. Van der Gaag, M. Henrion & F. Jensen, eds, 'Working Notes of the Workshop on Building Probabilistic Networks: Where Do the Numbers Come From?', pp. 23–31.

Kim, J. & Pearl, J. (1983), A computational model for causal and diagnostic reasoning in inference engines, *in* '8th International Joint Conference on Artificial Intelligence', Karlsruhe, West Germany, pp. 190–193.

Lin, J. (1991), 'Divergence measures based on the shannon entropy', *IEEE Trans. Inf. Theory* **37**, 145–151.

Miller, G. (1956), 'The magical number seven, plus or minus two: some limits on our capacity for processing information', *Psychological Review* **63**, 81–97.

Renooij, S. (2001), 'Probability elicitation for belief networks: issues to consider', *The Knowledge Engineering Review* **16:3**, 255–269.

Tang, Z. & McCabe, B. (2007), 'Developing complete conditional probability tables from fractional data for bayesian belief networks', *Journal of Computing in Civil Engineering* **21**(4), 265–276.

Van der Gaag, L. C., Renooij, S., Witteman, C., Aleman, B. & Taal, B. (1999), How to elicit many probabilities, *in* 'Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence 1999', pp. 647–654.

Wellman, M. P. (1990), 'Fundamental concepts of qualitative probabilistic networks', *Artificial Intelligence* **44**(3), 257–303.

Zagorecki, A. & Druzdzel, M. J. (2006), Knowledge engineering for Bayesian networks: How common are Noisy-MAX distributions in practice?, *in* G. Brewka, S. Coradeschi, A. Perini & P. Traverso, eds, 'ECAI 2006, 17th European Conference on Artificial Intelligence.', pp. 482–488.

Table 2: Performance of the three approximation methods on the measures specified in Section 3

	regarding CPT				regarding posteriors in scenarios				
Variable	av abs diff	av Je-Sh	max Je-Sh	unmatched 1/0	av abs diff	av Je-Sh	max Je-Sh	same likelh ranking	same most likely state
EBBN									
CombVerMo	0.074	0.028	0.099	0 (256)	0.034	0.002	0.025	93.6%	96.8%
CombMoisture	0.029	0.010	0.045	3 (64)	0.205	0.031	0.663	72.0%	82.4%
AreaMoDryAir	0.056	0.035	0.148	9 (64)	0.221	0.030	0.178	61.0%	82.0%
CombClouds	0.061	0.021	0.127	0 (27)	0.251	0.025	0.164	78.1%	82.8%
CldShadeOth	0.131	0.058	0.188	18 (144)	0.361	0.047	0.254	60.8%	70.9%
CldShadeConv	0.071	0.034	0.219	1 (36)	0.223	0.029	0.235	62.5%	73.8%
CompPlFcst	0.065	0.013	0.064	0 (324)	0.044	0.003	0.085	91.0%	92.4%
Copula vine									
CombVerMo	0.090	0.053	0.314	76 (256)	0.039	0.002	0.037	92.0%	95.2%
CombMoisture	0.075	0.040	0.153	7 (64)	0.274	0.037	0.607	60.0%	73.6%
AreaMoDryAir	0.053	0.023	0.072	16 (64)	0.195	0.019	0.094	61.0%	84.0%
CombClouds	0.105	0.043	0.133	1 (27)	0.315	0.036	0.240	76.6%	78.1%
CldShadeOth	0.103	0.040	0.127	23 (144)	0.279	0.032	0.265	78.1%	83.3%
CldShadeConv	0.056	0.015	0.067	2 (36)	0.157	0.012	0.079	71.2%	78.8%
CompPlFcst	0.143	0.069	0.408	0 (324)	0.085	0.012	0.428	86.9%	88.3%
Uniform									
CombVerMo	0.219	0.234	0.549	76 (256)	0.120	0.021	0.229	82.4%	82.4%
CombMoisture	0.130	0.117	0.415	7 (64)	0.797	0.205	0.549	23.2%	23.2%
AreaMoDryAir	0.238	0.273	0.520	16 (64)	0.819	0.218	0.524	25.0%	25.0%
CombClouds	0.289	0.199	0.408	1 (27)	0.810	0.188	0.445	28.1%	29.7%
CldShadeOth	0.293	0.225	0.459	23 (144)	0.747	0.175	0.550	26.2%	41.5%
CldShadeConv	0.149	0.092	0.250	2 (36)	0.404	0.072	0.274	31.2%	50.0%
CompPlFcst	0.142	0.063	0.253	0 (324)	0.089	0.011	0.315	83.4%	85.4%

underlined: best score for the three methods.

A Bayesian Approach to Learning in Fault Isolation

Hannes Wettig
Helsinki Institute for Information Technology
Finland
wettig@hiit.fi

Anna Pernestål
Dept. Electrical Engineering
Linköping University
Sweden
annap@isy.liu.se

Tomi Silander
Helsinki Institute for Information Technology
Finland
tsilande@hiit.fi

Mattias Nyberg
Scania CV AB
Södertälje
Sweden
mattias.nyberg@scania.com

Abstract

Fault isolation is the art of localizing faults in a process, given observations from it. To do this, a model describing the relation between faults and observations is needed. In this paper we focus on learning such models both from training data and from prior knowledge. There are several challenges in learning fault isolators. The number of data, as well as the available computing resources, are often limited and there may be previously unobserved fault patterns. To meet these challenges we take on a Bayesian approach. We compare five different methods for learning in fault isolation, and evaluate their performance on a real fault isolation problem; the diagnosis of an automotive engine.

1 INTRODUCTION

We consider the problem of fault isolation, i.e. the problem of localizing faults that are present in a process given observations from this process. To do this, a model of the relations between observations and faults is needed. In the current work we investigate and compare different methods for learning from training data and prior knowledge.

We are motivated by the problem of fault isolation in an automotive engine, and the learning methods are evaluated using experimental training data and evaluation data from real driving situations. In engine fault isolation there may be several hundreds of faults and observations. There will be fault patterns, i.e. co-occuring faults, from which there are no training data. Furthermore, training data is typically experimental and obtained by implementing faults, running the process, and collecting observations. On the other hand, there is often engineering knowledge available about the process. The engineering knowledge can for example be used to determine the structure of dependencies between faults and observations. This kind of knowledge is often the only basis in previous algorithms for fault isolation [6, 12, 19].

Due to the fact that there are previously unobserved fault patterns in training data, frequentist and purely data-based methods are bound to fail. To meet these challenges we use a Bayesian approach to learning in fault isolation. We consider five different methods of learning a model from training data, which are all previously present in the literature in different forms. We taylor these methods to incorporate the available background information. The methods we consider are Direct Inference (DI), Logistic Regression (LogR), Linear Regression (LinR), Naive Bayes (NB) and general Bayesian Networks (BN).

The main contributions of the current work are the investigation of Bayesian learning methods and regression models for fault isolation by comparing the five methods mentioned above, the application and evaluation of the methods on real-world data, and the combination of data-driven learning and prior knowledge within these methods. In order to do this investigation, we first discuss the characteristics of the fault isolation problem in terms of probability theory, and performance measures that are meaningful for fault isolation. Consecutively we show how the five methods can be adopted to the isolation problem. We apply them to the task of fault isolation in an automotive diesel engine. Finally, we compare the five methods, and discuss their advantages and drawbacks.

Bayesian methods for fault isolation are previously studied in literature. In these previous works it is generally assumed that the model is given [26, 15], or can be derived from a physical model without using training data [17, 25]. In the current work on the other hand, we focus on *learning* the models. Previous works on Learning models for fault isolation typically rely on pattern recognition methods described e.g. in

[1, 3]. Examples of such methods are presented for example in [14]. Pattern recognition methods are applicable if there is sufficient training data available. Unfortunately, this is rarely the case in fault isolation. In [20] the problem of learning with missing fault patterns is discussed. In [20] training data is combined with fundamental methods for fault isolation described in [2, 22]. This approach is referred to as Direct Inference in the current work, and compared to the other four methods for learning.

The paper is structured as follows. We introduce notation, and formulate the diagnosis problem in Section 2. Therein we also define relevant performance measures. In Section 3 we briefly describe the five methods used, and in particular how they are applied to the diagnosis problem, before we perform the evaluating experiments and compare the results obtained in Section 4. Finally, in Section 5 we conclude the paper by summarizing our results and discussing future work directions.

2 PROBLEM FORMULATION

Before going into the details of each of the learning methods we introduce some notation, and discuss the characteristics of the fault isolation problem. Then we carefully state the problem at hand and define performance measures.

2.1 BAYESIAN FAULT ISOLATION

The fault isolation problem can be formulated as a prediction problem, where the task is to determine the fault(s) present in a system, given a set of observations from the system. Let the faults be represented by the binary variables $\mathbf{Y} = (Y_1, \ldots, Y_K)$, and let the observations from the system be represented by the variables $\mathbf{X} = (X_1, \ldots, X_L)$, where each X_l is discrete or continuous. Generally, we use upper case letters to denote variables, and lower case letters to denote their values. Boldface letters denote vectors. We write $p(\mathbf{X} = \mathbf{x})$ (or simply $p(\mathbf{x})$) to denote either probabilities or probability distributions both in the continuous and in the discrete case. The meaning will be clear from the context.

We are given a set of training data $\mathcal{D}$, consisting of samples $(\mathbf{y}^n, \mathbf{x}^n)$, $n = 1, \ldots, N_{\mathcal{D}}$, pairs of fault and observation variables. The training data is collected by implementing faults and then collecting observations, meaning that training data is *experimental*. To evaluate the system we use a set $\mathcal{E}$ consisting of $N_{\mathcal{E}}$ samples. The evaluation data is collected by running the system, meaning that it is *observational*. Furthermore, we assume that the fault isolation algorithm is triggered by a fault detector telling us there must be *at least one fault present* in the process.

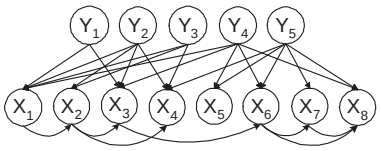


Figure 1: A Bayesian network describing a typical fault isolation problem.

The structure of dependencies between the faults and observations has three basic properties, illustrated in the example Bayesian network of Figure 1.

The first property is that faults assumed to be a priori independent, i.e. that

$$p(\mathbf{y}) = \prod_{k=1}^{K} p(y_k|y_1, \ldots, y_{k-1}) \approx \prod_{k=1}^{M} p(y_k), \quad (1)$$

meaning that faults cannot cause other faults to occur. Although not necessary for the methods in the current work, this is a standard assumption in many fault isolation algorithms [6], and it simplifies the reasoning in the following sections.

Second, faults may causally affect one or several of the observation variables introducing dependencies between faults and variables. A dependency between fault variable Y_k and observation variable X_l means that the fault *may* be visible in the observation.

The third property is that an observation variable X_l may be dependent on other observation variables. Dependencies between observation variables may arise due to several reasons. For example they can be caused by unobserved factors, such as humidity, driver behavior, and operation point of the process. These unobserved factors could be modeled using hidden nodes, but since they are numerous and unknown they are here simply modeled with dependencies between observation variables. This is more carefully discussed in [21].

We take a Bayesian view point on fault isolation. The objective is to find the probability for each fault to be present given the current observation, the training data, and the prior knowledge I, i.e. to compute the probabilities $p(y_k|\mathbf{x}, \mathcal{D}, I)$, $k = 1, \ldots, K$. The probability for each fault can be found by marginalizing over $\mathbf{y}_{-k} = (y_1, \ldots, y_{k-1}, y_{k+1}, \ldots, y_K)$,

$$p(y_k|\mathbf{x}, \mathcal{D}, I) = \sum_{\mathbf{y}_{-k}} p(\mathbf{y}_{-k}, y_k|\mathbf{x}, \mathcal{D}, I). \quad (2)$$

Note that $(\mathbf{y}_{-k}, y_k) = \mathbf{y}$, and (2) means that we seek the conditional distribution $p(\mathbf{y}|\mathbf{x}, \mathcal{D}, I)$. To simplify

the notation we will omit the background information I in the equations.

Computing the conditional distribution $p(\mathbf{y}|\mathbf{x}, \mathcal{D})$ is generally difficult. To approximate it we need a model $\mathcal{M}$ and a method for determining the parameters of the model.

2.2 PERFORMANCE MEASURES

To evaluate the different models to be used in Bayesian fault isolation, we use two performance measures: log-score and percentage of correct classification.

The log-loss is a commonly used measure [1], and given by

$$\mu(\mathcal{E}, \mathcal{M}) = \frac{1}{N_{\mathcal{E}}} \sum_{j=1}^{N_{\mathcal{E}}} \log p(\mathbf{y}^j|\mathbf{x}^j, \mathcal{M}), \qquad (3)$$

The scoring function μ measures two important properties of the fault isolation system; both the ability to assign large probability mass to faults that are present, and also the ability to assign small probability mass to faults that are not present. Furthermore, the log-score is a *proper score*. A proper score has the characteristic that it is maximized when the learned probability distribution corresponds to the empirically observed probabilities. In the fault isolation problem the conditional probabilities for faults is often combined with decision theoretic methods for troubleshooting [8], where optimal decision making requires conditional probabilities close to the generating distribution.

The second measure we use is not proper. It is closely related to the 0/1-loss used e.g. in pattern classification [1]. However, in case of multiple faults present it suffices to assign highest probability to any of them. We define

$$\nu(\mathcal{E}, \mathcal{M}) = \#\{j : y^j_{max}(\mathbf{x}^j, \mathcal{M}) = 1\}/N_{\mathcal{E}}, \qquad (4)$$

where $y^j_{max}(\mathbf{x}^j, \mathcal{M})$ is the fault assigned highest probability by $\mathcal{M}$ given $\mathbf{x}^j$. The ν-score reflects the performance of the fault isolation system combined with the simple troubleshooting strategy "check the most probable fault first".

3 MODELLING APPROACHES

In this section we briefly present the inference methods used to tackle the fault isolation problem. We carefully state all assumptions made, and describe the adjustments of each method to apply it to the diagnosis problem. However, we begin by describing two assumptions that need to be made for all methods except DI.

3.1 MODELLING ASSUMPTIONS

All the methods considered in this paper – with the exception of DI – build separate models for each fault and thus assume independence among these. A priori this corresponds to approximation (1). However, when we build separate models for each fault, we also make a stronger assumption, namely that the faults *remain* independent given the observations,

$$p(\mathbf{y}|\mathbf{x}) = \prod_{k=1}^{K} p(y_k|\mathbf{x}, y_1, \ldots, y_{k-1}) \approx \prod_{k=1}^{K} p(y_k|\mathbf{x}) \qquad (5)$$

This approximation is (after applying Bayes' rule and canceling terms) equivalent to

$$\prod_{k=1}^{K} p(\mathbf{x}|y_k) \approx \prod_{k=1}^{K} p(\mathbf{x}|y_1, \ldots, y_k), \qquad (6)$$

meaning that the observation $\mathbf{x}$ is dependent on each fault y_k, but this dependency is assumed to be independent of all other faults $y_{k'}, k' \neq k$. In other words, we assume *no "explaining away"* [10]. Looking at Figure 1 we observe, that this indeed is a strong assumption, since there are unshielded colliders (V-structures, bastards, common children of non-connected nodes) of the faults present.

Assumption (5) is primarily made for technical reasons, in order to be able to build separate models for each fault. But often it is also the case (as in the application of Section 4) that there is training data only from single faults. This means we do not have any training data telling us about the joint effect of multiple faults.

Remember that it is known that there is at least one fault present when the fault isolator is employed, see Section 2.1. Therefore, instead of computing $p(\mathbf{y}|\mathbf{x})$, we search

$$p(\mathbf{y}|\mathbf{x}, \sum_k y_k > 0) = p(\mathbf{y}|\mathbf{x})(1 - p(\mathbf{y} \equiv \mathbf{0}|\mathbf{x})). \qquad (7)$$

Unfortunately

$$p(\mathbf{y}|\mathbf{x}, \sum_k y_k > 0) \neq \prod_k p(y_k|\mathbf{x}, \sum_k y_k > 0), \qquad (8)$$

a fact which recouples the single-fault models introduced in (5). This fact is ignored during the learning phase and the single-fault models are trained individually. We then apply (7) in the evaluation phase.

3.2 DIRECT INFERENCE

Several previous fault isolation algorithms rely on prior knowledge about which observations may be affected

Table 1: An example of an FSM

	Y_1	Y_2	Y_3
X_1	1	1	0
X_2	1	0	1

by each fault [2, 22, 12]. Such information is typically expressed in a so called Fault Signature Matrix (FSM). An example of an FSM is given in Table 1. In the FSM, a zero in position (k, l) means that fault Y_k can never affect observation X_l. The direct inference method aims at combining the information given by the FSM with the training data available. Assume that observations are binary and that the background information I containing the FSM is given. Then, under certain assumptions it can be shown [20] that

$$p(\mathbf{y}|\mathbf{x}, \mathcal{D}) = \begin{cases} 0 & \mathbf{x} \in \gamma \\ \frac{n_{\mathbf{xy}}+\alpha_{\mathbf{xy}}}{N_{\mathbf{y}}+A_{\mathbf{y}}} \frac{p(\mathbf{y}|I)}{\pi_0} & \text{otherwise,} \end{cases} \quad (9)$$

where π_0 is a normalization constant, $n_{\mathbf{xy}}$ is the count of training data with fault $\mathbf{y}$ and observations $\mathbf{x}$, $\alpha_{\mathbf{xy}}$ is a parameter describing the prior belief in the observation $\mathbf{x}$ when the fault is $\mathbf{y}$ (a *Dirichlet* prior), $N_y = \sum_{\mathbf{x}'} n_{\mathbf{x}'\mathbf{y}}$, and $A_y = \sum_{\mathbf{x}'} \alpha_{\mathbf{x}'\mathbf{y}}$. The sets γ are determined by the background information as described in [20].

The direct inference method is developed for sparse sets of training data, particularly when there is only training data from a subset of the fault patterns to isolate.

3.3 BAYESIAN NETWORKS

When using Bayesian networks for prediction, we search the joint distribution $p(\mathbf{y}, \mathbf{x}|\theta)$, where θ are parameters describing the conditional probability distributions in the network. From the joint distribution, the conditional distribution for $\mathbf{y}$ can be computed. We consider two types of Bayesian networks: Naive Bayes and general Bayesian Networks.

3.3.1 Naive Bayes

The Naive Bayes classifier assumes that the observations are independent given the fault. Naive Bayes is is one of the standard methods for Bayesian prediction and often performs surprisingly well [3, 23]. However, due to the erroneous independence assumptions it is poorly calibrated when there are strong dependencies between the observations. To alleviate this problem, we apply variable selection according to an internal leave-one-out scoring function:

$$S(V) = \frac{1}{N_{\mathcal{D}}} \sum_{n=1}^{N_{\mathcal{D}}} \log P(y_k^n | \mathbf{x}^n, V, \mathcal{D} \setminus \{(\mathbf{y}^n, \mathbf{x}^n)\}, \alpha), \quad (10)$$

where $V \subset \mathbf{X}$ is the variable set under consideration and α is the Dirichlet hyper-parameter for the NB-model.

3.3.2 General Bayesian Network

Since it is known that the faults causally precede the observations, and since the observations are known to be dependent given the faults, a natural step forward from the Naive Bayes structure is a Bayesian network. In the network we constrain the fault to be a root node, but otherwise leave the structure unconstrained. One such network was learned for each fault using a BDe score (with an equivalent sample size parameter of 1.0). For small systems (< 30 variables) learning can be performed using the exact algorithm in [27], while for larger systems approximate methods, e.g. [9], can be used.

3.4 REGRESSION

Fault isolation is a discriminative task, where we are to predict the fault vector $\mathbf{y}$ given the observations $\mathbf{x}$, i.e. estimate the conditional likelihood

$$p(\mathbf{y}|\mathbf{x}, \theta) = \frac{p(\mathbf{y}, \mathbf{x}|\theta)}{\sum_{\mathbf{y}} p(\mathbf{y}, \mathbf{x}|\theta)}. \quad (11)$$

It is well known [18, 11] that in such case it can be of great benefit to employ a discriminative learning method, that only learns the probabilities asked, instead of wasting training data to learn the joint data likelihood as in the Bayesian network methods of Section 3.3. Regression models form a family of such methods.

3.4.1 Linear Regression

The most straight-forward regression method is linear regression, where each fault variable is assumed to be a linear combination of the observations plus a gaussian noise term,

$$y_k = \mathbf{w}_k^T \mathbf{x} + w_{k0} + \epsilon_k, \quad \epsilon \sim N(0, \sigma).$$

Here $\mathbf{w}_k$, w_{k0}, and σ are parameters to be determined. This gives the probability distribution

$$p(y_k|\mathbf{x}) = \frac{1}{Z} \exp(-\frac{(\mathbf{w}_k^T \mathbf{x} + w_{k0} - y_k)^2}{2\sigma^2}), \quad (12)$$

where Z is a normalization constant. To determine the parameters we use the standard methods described for example in [1].

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} - \sum_{n=1}^{N_\mathcal{D}} \log p(y_k^n|\mathbf{x}^n, \mathbf{w}) = \arg\min_{\mathbf{w}} - \sum_{n=1}^{N_\mathcal{D}} (\mathbf{w}_k^T \mathbf{x}^n + w_{k0} - y_k^n)^2.$$

When the parameters $\mathbf{w}^*$ are known, the parameter σ can also be computed. The normalization constant in (12) is given by $Z = \exp(-((\mathbf{w}^*)_k^T \mathbf{x} + w_{k0}^* - 1)^2/2\sigma^2) + \exp(-((\mathbf{w}^*)_k^T \mathbf{x} + w_{k0}^* - 0)^2/2\sigma^2)$.

3.4.2 Logistic Regression

Learning parameters to maximize (11) for a Bayes Net $\mathcal{B}$ is known to be equivalent to *logistic regression* under the condition that no child of the class can be a "bastard", a common child of two variables that are not interconnected directly. More formal definition and proofs can be found in [24]. In our case, this implies approximation (5).

To start with, for each fault we learn a logistic regression model corresponding to a discriminative Naive Bayes classifier [1].

We name the parameters of the logistic regression model α and β such that the conditional likelihood is defined as

$$p(y_k = 1|\mathbf{x}, \alpha, \beta) := \frac{\exp s(\mathbf{x}, \alpha, \beta)}{\exp s(\mathbf{x}, \alpha, \beta) + \exp -s(\mathbf{x}, \alpha, \beta)} \tag{13}$$

where

$$s(\mathbf{x}, \alpha, \beta) := \alpha + \sum_{l=1}^{L} x_l \beta_l. \tag{14}$$

We also include a smoothing term $c(\alpha, \beta)$ in our objective function which takes the place of a prior in the corresponding NB classifier. To unify its role for different observations, we first normalize our data by shifting and scaling such that for $l = 1, \ldots, L$

$$\sum_n x_l^n = 0 \quad \text{and} \quad \max_n |x_l^n| = 1 \tag{15}$$

Starting out from the uniform prior, we pretend to have seen one vector of each class at node Y_k and two vectors of each class with extreme values ± 1 at each node X_l, with all other values zero ($\sim$unobserved). This amounts to a smoothing term

$$c'(\alpha, \beta) - 2\log(\exp(\alpha) + \exp(-\alpha)) - 4\sum_{l=1}^{L} \log(\exp(\beta_l) + \exp(-\beta_l)). \tag{16}$$

However, we found this smoothing term problematic, since it is flat near zero. Therefore, we never get any parameters exactly zero. But in logistic regression many small parameters can make a difference, while they may be weakly supported. We choose to replace $\log(\exp(x) + \exp(-x))$ by $|x|$. This is a good approximation away from zero, but forces unsupported parameters to zero, implicitly performing attribute selection.

For fault Y_k we search parameters as to maximize

$$\log p(\mathbf{y}_k|\mathbf{x}, \alpha, \beta) + c(\alpha, \beta) = \sum_{n=1}^{N_\mathcal{D}} \log p(y_k^n|\mathbf{x}^n, \alpha, \beta) - 2|\alpha| - 4\sum_{l=1}^{L} |\beta_l|. \tag{17}$$

We do this by simple line search, one parameter at a time[2].

Finally, we try a variant of this algorithm which weights the training vectors. We have prior knowledge about the probabilities $p(y_k)$ with which to expect some fault y_k in the real-world setting or, in this case, the evaluation set. These probabilities differ from the relative frequencies observed in the training set. The idea is to weight the training vectors in the objective as to focus the optimization on areas of the data space more likely to be seen later on. The corresponding objective for fault Y_k becomes

$$\sum_{n=1}^{N_\mathcal{D}} \log w_k p(y_i^n|\mathbf{x}^n, \alpha, \beta) + c(\alpha, \beta) \tag{18}$$

where the weight w_k is the prior $p(y_k)$ divided by the observed relative frequency $\#\{n : y_k^n = y_k\}/N_\mathcal{D}$.

4 EXPERIMENTS

To evaluate the different methods learning fault isolation models, we apply them to the diagnosis of the gas flow in a 6-cylinder diesel engine in a Scania truck. In automotive engines, sensor faults are one of the most common faults, and here we consider five faults that may appear in different sensors. The faults are listed together with their prior probabilities in Table 2.

[1] possible other choices include tree-augmented Naive Bayes (TAN) [24, 5]

[2] There are much faster optimization techniques, some of which are compared in [16], but for our purposes this did nicely

Table 2: The faults considered

Fault	description	$p(y_k)$
y_1	exhaust gas pressure	0.4
y_2	intake pressure	0.13
y_3	intake air pressure	0.057
y_4	EGR vault position	0.13
y_5	mass flow	0.057

4.1 EXPERIMENTAL SETUP

For the gas flow of the diesel engine there is physical model from which a set of 29 diagnostic tests are automatically generated using structural analysis [4, 13]. Each of the observations is constructed to be sensitive to a subset of the faults.

For training and evaluation data we use measurements from real operation of the truck, with faults implemented. The training data consists of 100 samples each from the five single faults. Evaluation data consists of data from the five single faults, but also of data from two multiple faults $y_1\&y_2$, and $y_1\&y_4$. Evaluation data is observational, and consists of 1000 samples, distributed roughly according to the prior probabilities in Table 2.

The data we consider is originally continuous, but all except the regression algorithms take in discrete data. The data is discretized in two different ways: binary, with thresholds set such that all fault free data is known to be contained in the same bin; and discretized using k-means clustering [7] with $k = 4$. DI is applied to the discrete data. NB and BN are run both on discrete and binary data. The regression methods LinR and LogR are applied to the continuous data.

As described in Section 3 the NB and DI algorithms perform best if not all observations are used. For both DI and NB we perform variable selection such that an internal log-score is maximized. For DI, the best result is obtained by using only six of the observations. In NB between seven and 18 observations are used for each fault.

4.2 RESULTS

In Table 3 the log-score (μ) and percentage of correct classification (ν) are presented for the different methods. In addition we report the number of parameters used by each predictor. This is relevant, since for on-board fault isolation the computing and storage capacity is often limited. For comparison we also report the default which is obtained by simply using the prior probabilities given in Table 2.

Table 3: Comparison of the methods

method	log-score	ν-score	#pars
DI	-1.088	0.781	106
NB-bin.	-1.340	0.748	293
NB-disc.	-1.044	0.843	335
BN-bin.	-1.297	0.782	287
BN-disc.	-1.398	0.840	1136
LinR	-1.839	0.834	150
LogR	-1.071	0.829	46
LogR+weights	-0.953	0.829	44
default	-1.738	0.592	5

Table 4: Comparision of DI and LogR on single faults

fault	μ DI	μ LogR+w
y_1	-0.346	-0.385
y_2	-0.324	-0.287
y_3	-0.087	-0.008
y_4	-0.334	-0.294
y_5	-0.177	-0.133

We observe, that among the four best methods in Table 3 three are discriminative and learn the conditional distribution instead of the joint distribution. Furthermore, LogR with training sample weighting performs best on this data in log-score sense, while using a small number of parameters. Surprisingly the weighting trick has made quite a difference and LogR without weights it is outperformed by NB-disc. NB performs better when it is fed with discretized observations instead of binary, while for BN the effect is reversed. Clearly the discretized data contain more information, but it seems that in more complex Bayes Nets the conditional probability tables easily grow too large. In DI good results are obtained by exploiting prior knowledge in terms of that some faults never cause an observation to pass certain thresholds.

Measured by the ν-score the relative differences between the methods become smaller. We observe that this score favors the regression models and the Bayesian methods using binary data. The reason for the good performance of the methods using binary data is the particular way of thresholding the data such that all fault free samples are contained in the same bin.

Table 4 compares the log-scores of the predictions given for the single faults by DI and LogR+weights. Note that because of inequality (8) the columns do not sum to the corresponding entries in Table 3. Not surprisingly, both methods (as all others) have most

trouble with faults y_1, y_2 and y_4, the ones appearing simultaneously in evaluation data, but not in training data. This gives evidence for explaining away being important in this problem. Figure 2, in which the probabilities for each fault using LogR + weights are plotted, shows this in more detail. In the Figure we have ordered the evaluation data such that the rightmost samples have multiple faults, visualizing that the double faults are most difficult to predict.

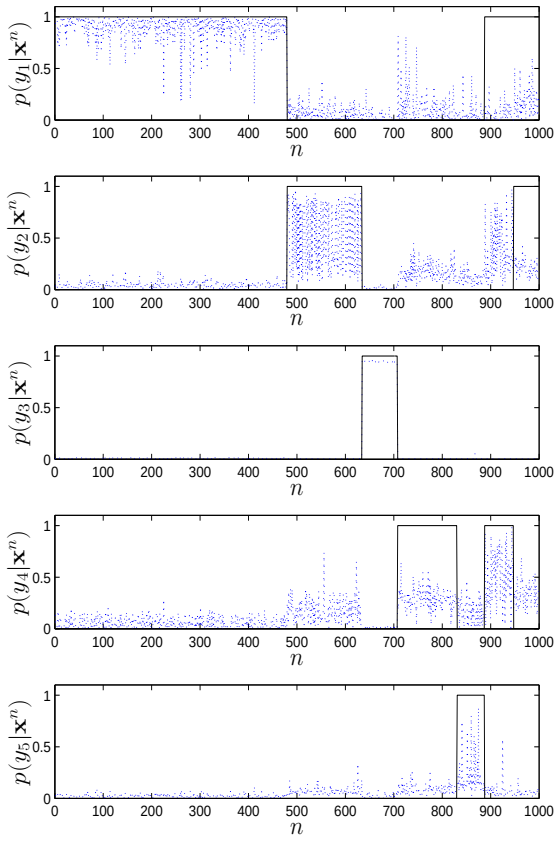


Figure 2: The predicted probability for the different faults given by LogR+w. Evaluation data is ordered after their fault patterns. The true fault is marked with a solid line.

5 CONCLUSIONS

We have considered the problem of fault isolation in an automotive diesel engine. We have discussed the special characteristics of this problem. There is experimental training data available which is distributed differently from what we expect to see in the real-world setting. In particular, evaluation data consists partly of previously unseen fault patterns. In addition there is prior knowledge available about which faults may affect each observation, and also the knowledge that at least one fault is present.

We have studied different Bayesian and regression approaches to combine this by nature heterogeneous information into probability distributions for the faults conditioned on given observations. We have compared the performance of the methods using real-world data, and have found that the discriminative logistic regression method to perform best. Among the best methods we have also found the naive Bayes classifier and the direct inference method.

One of the clearest implications of this work is that all methods have difficulties with handling unobserved fault patterns. Unfortunately, unobserved patterns are common in fault isolation, so this problem should be tackled in future work. All the methods used, except direct inference, ignore explaining away. However, this explaining away effect can possibly be helpful when diagnosing unseen patterns. Furthermore, it is crucial to include background information in the learning phase whenever it is available.

In our work to come we will investigate models capable of both explaining away and taking prior knowledge into account, while providing an efficient inference procedure, as on-board computers offer very limited resources. We expect further improvement of performance is possible.

References

[1] Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[2] Johan de Kleer and Brian C. Williams. Diagnosis with Behavioral Modes. In *Readings in Model-based Diagnosis*, pages 124–130, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[3] Luc Devroye, Laszlo Györfi, and Gabor Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, New York, 1996.

[4] Henrik Einarsson and Gustav Arrhenius. Automatic design of diagnosis systems using consistency based residuals. Master's thesis, Uppsala University, 2004.

[5] Russel Greiner and Wei Zhou. Structural Extension to Logistic Regression: Discriminative Parameter Learning of Belief Net Classifiers. In *13th international conference on uncertainty in artificial intelligence*, 2002.

[6] Walter Hamscher, Luca Console, and Johan deKleer. *Readings in Model-based Diagnosis*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1992.

[7] John A. Hartigan. *Clustering Algorithms*. Wiley, 1975.

[8] David Heckerman, John S. Breese, and Koos Rommelse. Decision-theoretic troubleshooting. *Communications of the ACM*, 38(3):49–57, 1995.

[9] David Heckerman, Dan Geiger, and David M. Chickering. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning*, 20(3):197–243, 1995.

[10] Finn V. Jensen. *Bayesian Networks*. Springer-Verlag, New York, 2001.

[11] Petri. Kontkanen, Petri. Myllymäki, and Henry. Tirri. Classifier learning with supervised marginal likelihood. In J. Breese and D. Koller, editors, *Proceedings of the 17th International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 277–284, 2001.

[12] Jozef Korbicz, Jan M. Koscielny, Zdzislaw Kowalczuk, and Wojciech Cholewa. *Fault Diagnosis. Models, Artificial Intelligence , Applications*. Springer, Berlin, Germany, 2004.

[13] Mattias Krysander, Jan Åslund, and Mattias Nyberg. An Efficient Algorithm for Finding Minimal Over-constrained Sub-systems for Model-based Diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 38(1):197–206, 2008.

[14] Gareth Lee, Parisa Bahri, Srinivas Shastri, and Anthony Zaknich. A multi-category decision support framework for the tennessee eastman problem. In *Proceedings of the European Control Conference 2007*, Greece, 2007.

[15] Uri Lerner, Ronald Parr, Daphne Koller, and Gautam Biswas. Bayesian Fault Detection and Diagnosis in Dynamic Systems. In *AAAI/IAAI*, pages 531–537, 2000.

[16] Thomas P. Minka. A comparison of numerical optimizers for logistic regression. Technical report, Micrsoft Research, 2003.

[17] Sriram Narasimhan and Gautam Biswas. Model-based Diagnosis of Hybrid Systems. *IEEE Trans. on Systems, Man, and Cybernetics, Part A*, 37(3):348–361, 2007.

[18] Andrew Y. Ng and Michael I. Jordan. On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes. In *Advances in Neural Information Processing Systems 14*, 2002.

[19] Mattias Nyberg. Model-Based Diagnosis of an Automotive Engine Using Several Types of Fault Models. *IEEE Transactions on Control Systems Technology*, 10(5):679–689, 2005.

[20] Anna Pernestål and Mattias Nyberg. Diagnosing Known and Unknown Faults from Incomplete Data. In *Proceedings of European Control Conference*, 2007.

[21] Anna Pernestål, Mattias Nyberg, and Bo Wahlberg. A Bayesian Approach to Fault Isolation with Application to Diesel Engine Diagnosis. In *Proceedings of 17th International Workshop on Principles of Diagnosis (DX 06)*, pages 211–218, 2006.

[22] Raymond Reiter. A Theory of Diagnosis From First Principles. In *Readings in Model-based Diagnosis*, pages 29–48, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[23] Irina Rish. An empirical study of the naive bayes classifier. In *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, 2001.

[24] Teemu Roos, Hannes Wettig, Peter Grünwald, Petri Myllymäki, and Henry Tirri. On Discriminative Bayesian Network Classifiers and Logistic Regression. *Machine Learning*, pages 267–296, 2005.

[25] Indranil Roychoudhury, Gautam Biswas, and Xenofon Koutsoukos. A Bayesian Approach to Efficient Diagnosis of Incipient Faults. In *Proceedings of 17th International Workshop on Principles of Diagnosis (DX 06)*, pages 243–250, 2006.

[26] Matthew Schwall and Christian Gerdes. A probabilistic Approach to Residual Processing for Vehicle Fault Detection. In *Proceedings of the 2002 ACC*, pages 2552–2557, 2002.

[27] Tomi Silander and Petri Myllymäki. A Simple Approach for Finding the Globally Optimal Bayesian Network Structure. In *Proceedings of the 22nd Conference on Uncertainty in AI (UAI)*, 2006.

Hypothesis Management Framework: a flexible design pattern for belief networks in decision support systems

Sicco Pier van Gosliga, Imelda van de Voorde
TNO Defence, Security and Safety
The Hague, The Netherlands

Abstract

This article discusses a design pattern for building belief networks for application domains in which causal models are hard to construct. In this approach we pursue a modular belief network structure that is easily extended by the users themselves, while remaining reliable for decision support. The Hypothesis Management Framework proposed here is a pragmatic attempt to enable analysts and domain experts to construct and maintain a belief network that can be used to support decision making, without requiring advanced knowledge engineering skills.

1 INTRODUCTION

Since their introduction by Kim and Pearl [10] belief networks have become a popular framework for decision support and automated reasoning. Also at TNO, the Netherlands Organisation for Applied Scientific Research, Bayesian reasoning is used in an increasing number of projects and application domains. One of these application domains is decision support for criminal investigations. The typical application in this field is to perform a quick scan on available evidence to select the most likely hypothesis, and to prioritize unavailable evidence to aid further investigations. The need for sound probabilistic reasoning is quite large in this area, and belief networks are becoming an accepted tool for modeling reasoning.

Well-known examples of belief networks such as the Alarm [2] and Hailfinder [1] networks are quite complex and their development requires the co-operation between both Bayesian specialists and domain experts. Also, currently available software packages (e.g. HUGIN, Netica and GeNie)[1] for modeling and analysing belief networks require expertise and skill in belief networks. Whereas in the field of criminal investigations, the typical user of such decision support software is usually not a Bayesian specialist but either an analyst or an expert on the area being analyzed, a so-called domain expert. To get belief networks accepted as a standard tool in criminal investigations, we should improve the usability to such a degree that a domain expert is able to produce useful models without the assistance of a Bayesian specialist. Obviously, analysts should find it beneficial for performing their analyses as well.

Besides offering criminal investigators a method to use belief networks, also some effort should be focused on preventing bias arising in analyses. Where much attention goes into getting unbiased and accurate prior probabilities, in this paper we are more concerned with any bias within the topology; the choice of variables included in the model. When an analyst looks for support for a certain hypothesis, it is easy to get into a so-called tunnel view in which contradicting evidence and alternate hypotheses are neglected. When a plausible alternative perspective is missing in the model, a potential bias is present yet invisible. It seems impossible to always exclude such a bias, but applying certain strategies in the design of a belief network may lead to more balanced and less biased models. Among others, the following strategies might be considered. Firstly, different domain experts can add an alternative point of view to the same model. Secondly, each domain expert can work independently on a different hypothesis or counter-hypothesis. And finally, domain experts can design reusable templates that are not tailored for a specific case, but for generic classes of cases. Whatever combination of strategies may work best to avoid a bias, the case for a flexible and modular way

[1] The software packages HUGIN Expert, Netica and GeNie are respectively found at: `http://www.hugin.com`, `http://www.norsys.com/netica.html` and `http://genie.sis.pitt.edu/`

to design belief networks to aid better decision making should be apparent.

Various systematic techniques are available to guide the modeling of a belief network in a systematic manner. Many of these generate a belief network by translation of another type of model, e.g. ontologies [19], rule-based systems [11], causal maps [16], or by merging quantitative and qualitative statements in a canonical form [5]. However, all these techniques rely on a sound understanding of the application domain to establish the qualitative aspect of a belief network: the topology of the graph. When a domain is modeled that is dynamic in nature and of which causality is not fully known, the technique used to construct a belief network must above all be modular and easily extendible as new insights constantly change the perspective of what variables matter to the hypotheses of interest.

This led to the development of the hypothesis management framework (HMF) at TNO. This design pattern enables a domain expert to independently create and maintain a belief network, and an analyst to evaluate evidence in a criminal investigation. The HMF is a modular belief network structure that is easily expandable by the users themselves, while remaining reliable for decision support. The HMF adds a layer of abstraction to the belief network, so the belief network can be kept hidden from the user. Multiple users can independently modify or extend the model based on his or her domain knowledge. The HMF ensures that all parts of the model remain a coherent whole, suitable for consistent reasoning.

2 THE PURPOSE OF HMF

While devising the HMF design pattern we had one particular goal in mind: to enable the design of modular and extendible Bayesian models for users that are no Bayesian specialist. Once a first version of a model has been developed, it should be easily extended and maintained later-on. It is likely that the set of variables as well as the subjective priors for conditional probability tables require regular revisions as the field of investigation changes over time. Therefore it should be possible to reconsider the set of variables, without having to elicit all of the priors on each change of the model. The need for multiple revisions of a developing model was addressed by the AI group at the University of Kentucky in [14]. A design pattern should preferably be such that it enables the use of templates, generalized submodels within the belief network, that can be maintained independently by a group of domain experts. Such templates should be applicable within multiple belief networks.

To maximize its applicability in real world applications the following two requirements should be met:

1 *Reliability (or consistency)* The belief network should capture the knowledge of domain experts. Given the same set of evidence, the domain experts should agree on the same most likely hypotheses and the results of the model should intuitively make sense.

2 *Usability* The number of priors to be elicited should be kept to a practical minimum. We prefer to have a limited set of well founded priors, rather than a larger set of priors of which the domain expert is less confident. Conditional probability tables with a small set of priors are easier to maintain and validate, especially when the number of conditioning parent variables is limited. Furthermore, it should be unambiguous to domain experts (as well as the analysts) what the variables and their priors stand for.

These requirements are indeed very common, and generally accepted as basic requirements in the context of system development. We think, however, that they are hard to comply with without the use of a generalized framework.

3 AN OVERVIEW OF HMF

The HMF places each variable of interest within a predefined structure, as visualized in Figure 4(c). Furthermore it prescribes which variables may be instantiated with evidence, and for some variables the content of conditional probability tables. All variables must be categorized by the user in hypotheses, indicators or information sources. Each type has its own place and role within the topology of the belief network:

1 *Hypotheses* are statements of which we would like to get a posterior probability distribution. In general, hypotheses are unobserved. The user can specify unconditional priors for each hypothesis, or use a uniform nondiscriminative distribution instead. As an option, one can add alternative hypotheses to represent known facts that explain observed indicators in an other way than existing hypotheses.

2 *Indicators* are statements related to hypotheses. Knowledge of an indicator helps to reveal the states of related hypotheses. Indicators describe events that are dependent on the occurrence of one or more hypotheses. Causal relations between hypotheses and indicators are not always obvious, or present at all. Indicators are assumed to be

'caused' by hypotheses, not the other way around. For each relation between an indicator and a hypothesis, a domain expert should specify conditional probabilities for that specific relation.

3 *Information Sources* are used to express the reliability of sources related to an indicator, when the user does not want to enter 'hard evidence'. For instance, an information source may be a report, a sensor or a person. An indicator can be associated to multiple information sources.

Although common, it is not necessary for an arc in a belief network to imply causality. The HMF makes use of this freedom by taking a more abstract perspective on the relations between variables of interest. The structure is based on the relatively simple notion of hypotheses and indicators. Indicators may all support or contradict any of the hypotheses, but the indicators themselves are assumed independent of one another. Hypotheses are independent (root nodes) and typically have many children. Quite similar, so-called 'naive Bayes' structures [6], have been effective in other areas where causality is unknown or too dynamic in nature (e.g. e-mail spam filtering [15]).

If more structure is desired, this modeling style may be applied in a recursive fashion in which a hypothesis may have sub-hypotheses, who are modeled in an similar way. This is not demonstrated in this article.

It is good practice to use a causal model whenever possible [17], and it should be stressed that HMF does not aim to substitute such models. The HMF design pattern is specifically designed for domains in which causal dependencies are debated or not fully known. As pointed out by Biedermann and Taroni [3], in forensic science the availability of hard numerical data is not a necessary requirement for quantifying belief networks and Bayesian inference could therefore be used nonetheless. By using HMF, a Bayesian model can be constructed even when the qualitative aspects of a belief network are hard to obtain.

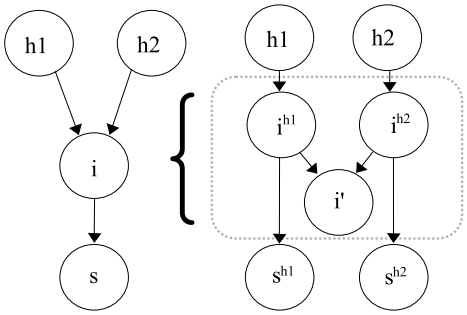


Figure 1: Indicators are substituted by multiple intermediate variables and one combining variable.

There are various options to elicit priors for such large CPTs. One could apply linear interpolation over a subset of elicited priors [20], but this requires more elicited priors and is less flexible than the solution found for HMF. Rather than connecting indicators directly to hypotheses, the HMF uses intermediate variables. In this article all variables are booleans. This is not a strict requirement, but a general recommendation when it simplifies the elicitation of prior probabilities. Elicited priors will be stored in the intermediate variable between the indicator and the hypothesis. This reduces the number of prior probabilities to elicit, and conditions to consider for each prior. In fact, the HMF splits up each indicator in multiple variables (Figure 1): one or more intermediate variables (i^{h1}, i^{h2}) and a variable that combines them (i'). For three hypotheses i would require 16 priors, instead of 12 priors for the three intermediate variables together.

When evidence is available for an indicator, we instantiate all associated intermediate variables. Alternatively, one can use information sources. An information source for an indicator (s in Figure 1) may exist as multiple variables with identical priors in the HMF belief network (s^{h1}, s^{h2}). The priors of an information source variable represent the reliability of the source in regard the associated indicator. Information source variables are children of intermediate variables, and have only one parent and no children. Either all information sources of an indicator are instantiated for evidence, or all associated information source variables. Instantiating intermediate variables of an indicator d-separates information sources from hypotheses, rendering all information sources for that indicator obsolete.

When there is no evidence for an indicator, the combining indicator variable (i') will resemble the posterior probability of the original indicator (i) by taking the average probability of all intermediate variables. This information is useful to predict the likelihood of unobserved indicators or for selecting the most influential unobserved indicator. Equation 1 is used to construct the conditional probability table of the combining indicator variable. Note, that the HMF does not use a logical function (e.g. OR/MAX, AND/MIN or XOR). Logical functions that assume independence of causal influence, in a discrete or noisy variant, have been long in use [8] as a solution for variables with many parents. An extensive overview of such methods are described by Diez and Drudzel in [4]. Although many alternatives may be considered, our preference goes to an averaging method to avoid scalability problems. The scalability problem will be further discussed in Section 5, while the results of using the averaging method in Equation 1 are discussed in Section 6.

$$P(X|parents(X)) = 1.0 - \frac{maxValueOf(parents(X))}{valueOf(parents(X))} \quad (1)$$

This article focuses on how HMF can aid the construction of belief networks. It does not elaborate on how a software tool might facilitate this process. Nonetheless, we would like to discuss briefly how we envision such a tool and how the HMF might be presented to the user. We differentiate two types of roles for users: domain experts and analysts. A user may have both roles in practice. By using the HMF, the GUI can effectively hide the underlying belief network from the user. Both types of users need a different user interface.

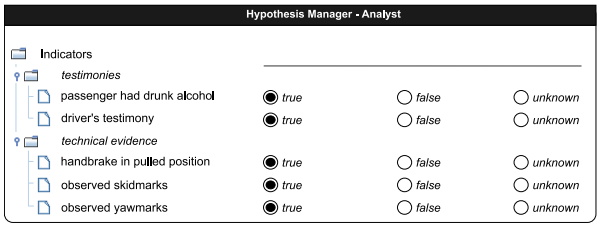


Figure 2: The GUI for an analyst.

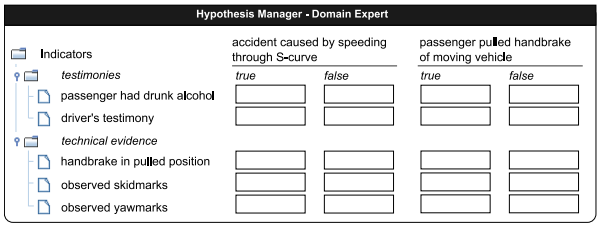


Figure 3: The GUI for a domain expert.

An analyst processes information sources and selects evidence for indicators to support or contradict hypotheses. For analysts the GUI (Figure 2) shows indicators in a foldable tree-like structure. The indicators are organized in categories and sub-categories. For each indicator the analyst can choose a state (e.g. true or false) based on observed evidence. If the analyst is uncertain about an observation, the analyst is given the ability the express the reliability of each information source for that specific indicator. This requires a prior probability for both positive observations and false positives, given that the indicator is a boolean.

Domain experts evaluate the conditional probabilities of an indicator given an hypothesis, and choose prior probabilities for hypotheses. The GUI should enable a domain expert to construct and maintain a list of indicators and hypotheses. A domain expert is responsible for relating indicators to hypotheses in a sensible manner, and assign conditional probabilities to each relation. Figure 3 shows how this may be presented to the domain expert. There is a column for each hypothesis. Assuming only booleans are used, the respective column requires only two elicited priors: one prior for the likelihood of observing the indicator given the hypothesis is true, and another for when the hypothesis is false. Qualitative descriptions or frequencies can be more effective than probabilities [7]. Such notations can be used instead of probabilities, as long these descriptions are consistently translated into conditional probability tables.

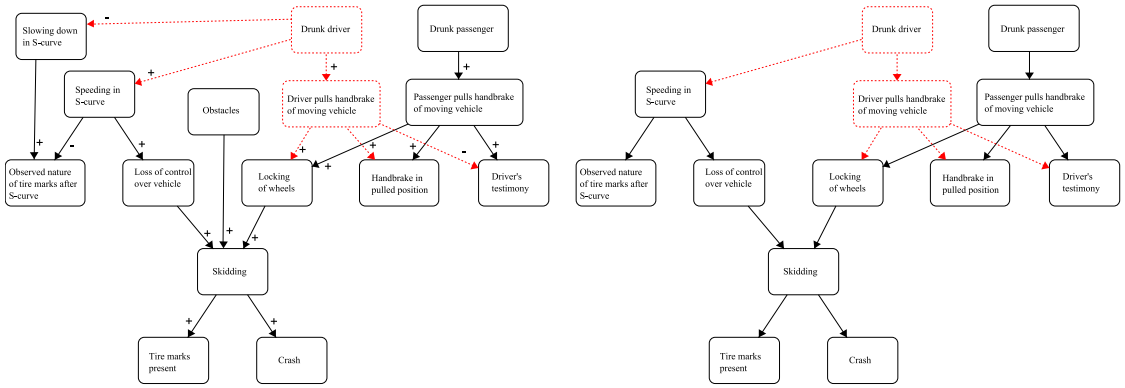
4 HMF WALKTHROUGH BY AN EXAMPLE

To explain how the HMF may be used and why we have chosen this specific topology, we will now discuss three different models based on a civil case concerning a car accident. The first is a logical causal model by Prakken and Renooij [18]. The second is a Bayesian belief network by Huygen [9], directly based on Prakken's logical model. Third and finally, a Bayesian belief network that follows the HMF is constructed for the same case. [2]

The legal case concerns a nightly car accident involving a driver and a passenger, after a party which both persons attended. The police that arrived at the scene after the accident observed that the car crashed just beyond an S-curve and the handbrake was in a pulled position. The police did observe tire marks (skid marks and jaw marks), but did not observe any obstacles. The driver claims that the passenger was drunk and pulled the handbrake. The passenger claims that the driver speeded through the S-curve. The judge had to decide whether it is plausible that the passenger caused the accident, rather than the driver.

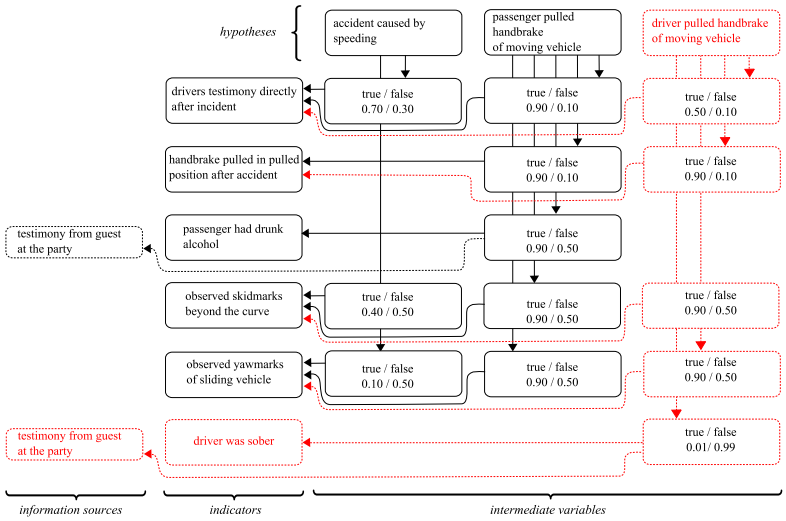
The logical model about this case by Prakken and Renooij is aimed at reconstructing the reasoning behind the court decision on this case. Figure 4(a) shows the causal structure for the case. Nodes within the structure visualize causal concepts (propositions), and arcs represent causal rules between them. Each arc is annotated to show whether the proposition at the head supports (+) or contradicts (-) the proposition at the tail of the arc. By using abductive-logical reasoning on the structure given evidence for some concepts, one can determine whether other concepts are plausible. Although such a model, a causal map, like the one in Figure 4a may resemble a belief network, it lacks the quantitative information required for Bayesian inference. Nadkarni and Shenoy [16] discussed how a causal

[2] the belief networks discussed in this article are available for download at: `http://www.science.uva.nl/~spg`



(a) A logical causal model by Prakken and Renooij.

(b) The belief network by Huygen.



(c) A variant that uses the HMF design pattern.

Figure 4: Three different models of the same case. The colour red is used to highlight the proposed extensions.

map, can be used as a foundation for constructing belief networks when supplemented with casual values that express the strength of a causal connection.

There is evidence for the following facts: $\neg$*obstacles*, *tire marks present*, *observed nature of tire marks after S-curve*, *handbrake in pulled position*, *driver's testimony* and *drunk passenger*. The hypotheses *speeding in S-curve* and *loss of control over vehicle* explain two facts but contradicts three others. Whereas the hypothesis *passenger pulled handbrake of moving vehicle* explains three rules and contradicts nothing. This makes the drivers point of view more convincing.

Huygen used the causal model of Prakken to construct a belief network for the same case (Figure 4b). The topology was slightly changed: the node for *obstacles* has been removed and the propositions for *speeding* and *slowing down in S-curve* have been replaced by a single boolean that represents both. Furthermore, each node is accompanied with a conditional probability table or prior probability distribution (not visible in Figure 4(b)). This effectively replaces the annotations along arcs in the causal map. Huygen decided not to use evidence for variables on tire marks, because in the sentence of the court it was not explicitly stated that the nature of the tire marks were proof for not speeding, but gave insufficient support for the suggestion that the driver had speeded. Huygens suggests to change the priors, when one would like to use this evidence.

Given evidence for: *pulled position*, *driver's testimony*, *passenger drunk* and *crash*, it is highly likely that the passenger pulled the handbrake ($\approx 100\%$). Since the evidence against the passenger explains away the car crash, it is unlikely that the crash was caused by lost control of the vehicle after speeding through the S-curve (0.1%). The bayesian belief network comes to the same conclusion as the causal map of Prakken and Renooij.

When we model the same case using the HMF, we get a radically different topology (Figure 4(c)) that does

not resemble the causal map of Prakken and the belief network of Huygen. Both claims are modeled as hypotheses in the HMF model: *accident caused by speeding* and *passenger pulled handbrake of moving vehicle*. These hypotheses correspond to similarly named predicates in Figure 4a and probability variables in Figure 4b. Uniform probability distributions were used as priors for these hypotheses. We use indicators to support our beliefs in the hypotheses, these are: *driver's testimony directly after incident*, *handbrake in pulled position after incident*, *passenger had drunk alcohol*, *observed yawmarks of sliding vehicle* and *observed skidmarks beyond the curve*.

By choosing different priors, the evidence for tire marks is now usable. Some intermediate variables that relate facts with the two hypotheses are no longer in use. These are *locking of wheels* and *loss of control over vehicle*. The information source of *passenger had drunk alcohol* is undisclosed. Suppose the source was a guest at the party, than the reliability of this testimony is represented by an information source variable (Figure 4(c)).

Given the available evidence, we get a high likelihood for the passenger pulling the handbrake of the moving vehicle ($\approx 100\%$). The propability for speeding is much lower ($\approx 27\%$), and therefore far less convincing.

All three approaches can adequately model the case and derive equally sensible conclusions. Abductive-logical reasoning over a causal map explains the logical correctness and contradictions of propositions. The advantage of a Bayesian approach is that by quantifying influence, it is able to give insight in what hypothesis is most credible as well as the relevance of evidence. The models of Prakken, Renooij and Huygen are based on a causal map. Although HMF follows a different approach to the construction of belief networks, and therefore uses a rather different topology, it does derive the same conclusions.

5 ISSUES REGARDING EXTENDIBILITY

Extendibility as well as modularity are important requirements. The models by Prakken and Huygen are 'static' models in the sense that they were designed to model one single case with a fixed set of evidence and hypotheses. This is feasible when consensus has been developed on all aspects of the case. However, supporting decision making at an earlier stage requires a high level of flexibility. The HMF was developed to facilitate decision making when the set of evidence (or indicators) and hypotheses is still evolving and a constant topic of discussion. Models designed with the HMF are flexible, meaning that a model is decomposable into independent modules. So that each module can be maintained or extended by a different domain expert. This section will discuss issues that concern the extendibility of models developed with the HMF. These issues will be illustrated by extending the existing models from the previous section.

We have pursued extendibility by modular independence of the elicited priors. When an indicator is added to the model, the only priors to elicit are those for the intermediate nodes of that specific indicator. Priors that were elicited before do not have to be reconsidered. The same holds for adding hypotheses. We will illustrate this by considering an additional hypothesis for the car accident case. Suppose the *driver* pulled the handbrake of the moving vehicle. If the driver was under influence of alcohol, that would have also influenced the driving behavior and therefore the likelihood of speeding as well as the possibility of pulling the handbrake of the moving vehicle. In all three models we would have to add and update existing prior knowledge.

To add the alternative hypothesis to the logical model of Prakken and Renooij a proposition is needed for the new hypothesis, and another to represent the possibility that the driver was under the influence of alcohol. These additional causal relations are highlighted in red in Figure 4(a). Together, these additions extend the existing set of 12 rules with 6 more.

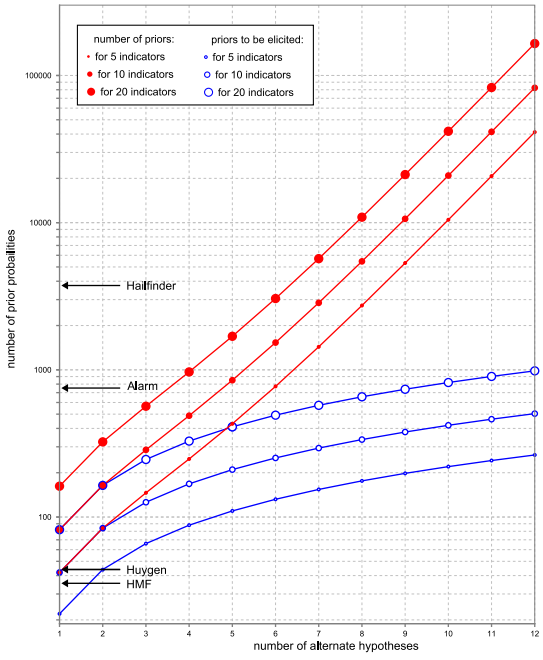


Figure 5: How extending the model affects the number of priors to elicit.

Table 1: Extending the models.

priors	Prakken	Huygen	elicited HMF
in original model	12	44	36
after extension	18	64	58
unchanged	12	30	36
updated and added	6	34	22
relative workload	50%	77%	61%

When we add similar variables and relations to the belief network of Huygen, we need to specify new conditional probability tables for *locking of wheels*, *handbrake in pulled position* and *driver's testimony*. Furthermore, we would have to replace the prior probability distributions of *speeding through S-curve* with a new conditional probability table. These changes comprise the elicitation of 34 new priors that substitute 14 previously elicited priors.

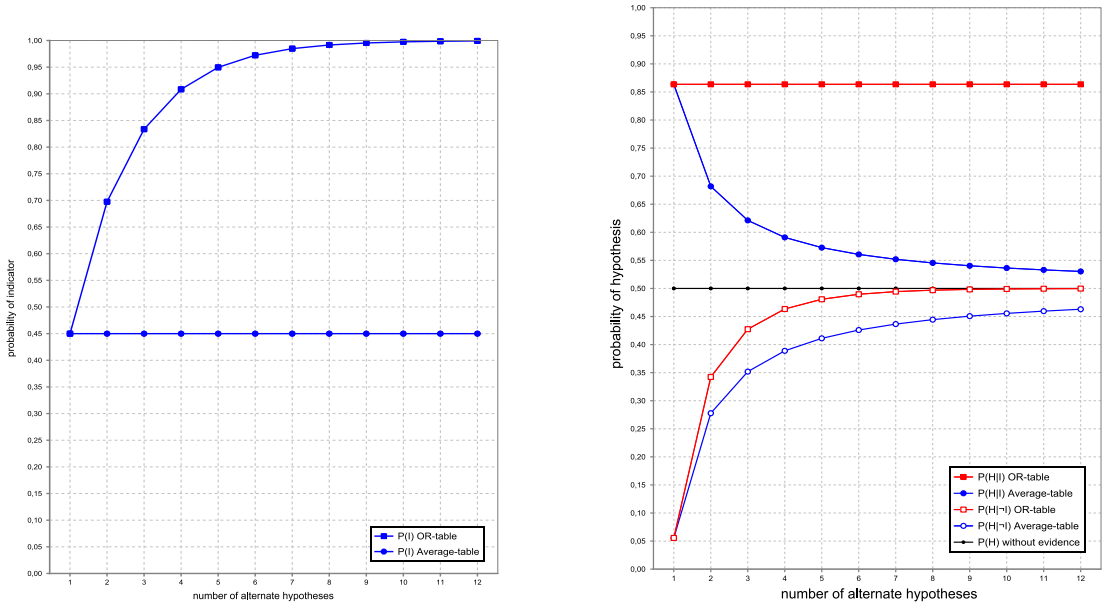
To add to the HMF model the hypothesis *driver pulled the handbrake of the moving vehicle*, requires a new column in the model in Figure 4. The possibility of the driver being under the influence of alcohol is modeled as an indicator, which adds a new row to the model. Table I shows how many elicited priors are required for extending the models. The extensions of the HMF model comprise only 22 elicited priors, all 36 existing priors remain unchanged. This makes HMF considerably cheaper to extend than the belief network of Huygen. The original causal model of Prakken is even simpler to extend. That model, however, lacks quantitative support for probabilistic inference.

As the car accident case shows, the HMF is tolerant to extensions. Figure 5 shows the general effect of adding hypotheses and indicators to a model by outlining the maximum number of elicited priors. While the total number of parameters grows exponentially when more hypotheses are added, the amount of elicited priors grows in a linear fashion. The figure assumes the worst case in which each indicator is associated to all hypotheses. Although the model assumes boolean variables and two priors for each intermediate variable would suffice, it is assumed that *all* priors for intermediate variables are elicited as well as a prior probability distribution for each hypothesis. Note that we have excluded all other parameters that require elicitation such as variable names and state definitions. As a reference Figure 5 includes the number of priors of Hailfinder (3741), Alarm (752), the original belief network of Huygen (44) and the HMF model from Section 4 (36). The extensions proposed in this Section were excluded from the HMF model.

As mentioned in Section 3 indicators are modeled by intermediate variables and one combining variable. The more hypotheses are associated to an indicator, the more probabilities of intermediate variables will have to be combined. On each extension the combining variable gets an extra parent, and as a consequence its conditional probability table (CPT) doubles in size. In the HMF an averaging function has been chosen as the preferred option for these CPTs. By default, the CPT of a combining variable effectively takes the average posterior distribution of all intermediate variables (Equation 1).

Arguably, one might find a logical OR-function [8] more intuitive. However, we have chosen not to use an OR or AND function for these CPTs since a methodical bias may arise in the model if it is extended. A practical drawback of using OR-tables in this situation arises when more than (approximately) five alternative hypotheses are connected to an indicator. By adding more parents to a deterministic OR-table the probability for the child variable quickly converges to unity, or alternatively a pre-defined upper bound. This is shown in Figure 6(a). It is likely that this will lead to unintentional overestimation of the occurrence of unobserved indicators. This can be illustrated by extending the belief network of Huygen, where the variable *locking of wheels* is modeled as an OR-table with an upper bound of 0.80. Suppose the case would be extended to include one or two additional drunk backseat passengers who may have pulled the handbrake of the moving vehicle. The extra backseat passengers are modeled in the same way as the passenger in front, using the original priors $P(locking|pulled) = 80\%$ and $P(locking|\neg pulled) = 0\%$ (where pulled is *true* when any of the persons in the vehicle pulled the handbrake). Given that the driver is sober and all passengers are drunk, the probability of *locking the wheels* increases rapidly (one drunk passenger: 2.4%, two drunk passengers: 4.7%, three drunk passengers: 7.0%). Even when we have not instantiated any other variables (e.g. *crash* or *driver's testimony*). After these extensions, one might like to reconsider the original priors of $P(pull|drunk)$ to prevent overestimating the probability of locked wheels. This potential problem is avoided when the method in Equation 1 is used.

Another potential problem that is associated with OR-tables is the asymmetric influence of an indicator: positive observations have less impact than a negative observation. This is shown in Figure 6(b)). Where observed indicators will only have marginal impact on hypotheses when observed true, the impact on intermediate variables of an indicator observed as *false* is deterministic and therefore usually stronger. It is likely that the user will be unaware of these effects. This makes the model relatively vulnerable to errors in the priors. Therefore, we advice to use Equation 1 as the default method. Other methods for constructing CPTs of combining variables may hinder extending



(a) The likelihood of an indicator.

(b) The impact of evidence for an indicator.

Figure 6: Extending the model affects the probabilities.

the model.

6 ISSUES REGARDING RELIABILITY

To evaluate the outcomes of HMF belief networks we have translated the Asia belief network, as introduced by Lauritzen and Spiegelhalter in [12], into the HMF format.

We will use abbreviations that correspond to the first character of each variable. The original model is shown in Figure 7 (left), the HMF version of Asia is shown on the right. In the HMF model of Asia we distinguish hypotheses: $\{b, l, t\}$, indicators: $\{s, v, x, d\}$ and intermediate nodes: $\{s_b, s_l, v_t, x_b, x_l, x_t, d_b, d_l, d_t\}$. The variable $TbOrCa$ is missing from the HMF model, which in the original belief network combines the probabilities of tuberculosis and lung cancer with a logical OR function has become obsolete.

In the HMF model of Asia, the prior information for the indicators is specified separately for each associated hypothesis. This assumes that the influence of e.g. lung cancer on dyspnea is unaffected by bronchitis. The following probabilities will have to be elicited from a domain expert, when using HMF on Asia. Unconditional priors for each hypothesis: $P(b)$, $P(l)$, $P(t)$ and conditional priors for all intermediate nodes: $P(s_b|b)$, $P(s_l|l)$, $P(v_t|t)$, $P(x_b|b)$, $P(x_l|l)$, $P(x_t|t)$, $P(d_b|b)$, $P(d_l|l)$, $P(d_t|t)$.

The Asia model uses only boolean variables and therefore only one probability for each hypothesis has to be elicited and two for each association of an indicator with a hypothesis. For Asia this gives a total of 21 probabilities. In this case the priors for the hypotheses and intermediate nodes were derived from the joint probability table of the original Asia belief network.

We computed the posteriors of the hypotheses for all possible scenario's of evidence for the indicators. In each of these scenarios each indicator was either observed or not. Note that we instantiate the intermediate nodes for evidence, rather than the combining variables. As mentioned in Section 3 an indicator is represented by both intermediate variables and a combining variable. The conditional probability table of the combining variable is implemented by Equation 1, whereas the elicited priors are stored in the intermediate variables. Instantiating only the combining variable would undervalue those elicited priors.

The results are shown in Table II. For each indicator and hypothesis, the table shows the average and maximum absolute difference in posteriors, as well as the Jensen-Shannon divergence [13]. The bottom row shows the percentage of scenario's in which the outcomes (i.e. the most likely state) for the variables were equal. Especially this last criterion is important for decision making, as the 'real' priors and posteriors will always be open to debate when a causal model is hard to obtain. The table shows that while posterior distributions may vary between both versions, on average the difference is relatively small (< 4 percentage points). For almost all scenario's the outcomes

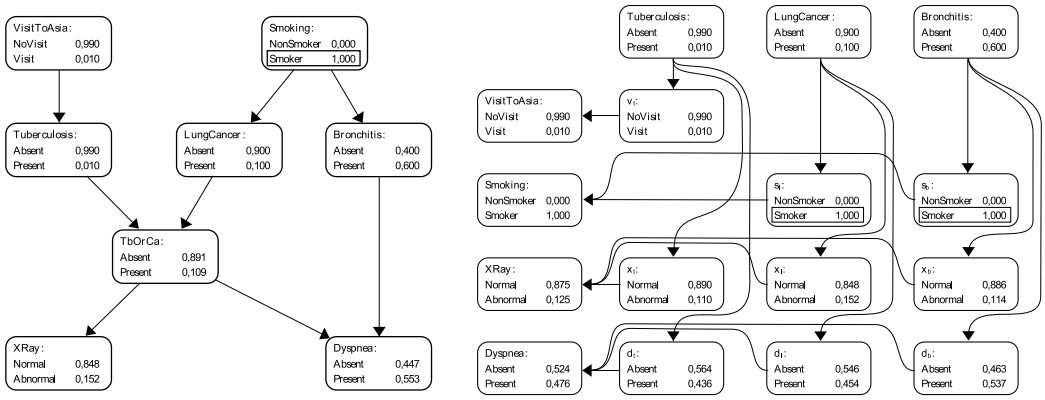


Figure 7: Left: the original Asia belief network. Right: HMF version of Asia. Both with evidence for *Smoking*.

Table 2: Divergence between HMF verion of Asia and the original.

vertice	d	v	x	b	t	c	s
max dif	0,162	0,004	0,071	0,308	0,193	0,209	0,095
av. dif	0,023	0,000	0,009	0,036	0,020	0,017	0,014
max J-S	0,021	0,000	0,006	0,074	0,029	0,035	0,008
av. J-S	0,002	0,000	0,001	0,006	0,002	0,002	0,001
match(%)	91,4	100,0	100,0	97,5	98,8	98,8	97,5

are identical. The few exceptions are caused by the synergistic effect between an abnormal X-Ray and the presence of dyspnea. This synergistic affect is absent in the HMF version, and in those situations we get the relatively large differences in the posterior distributions of bronchitis and long cancer.

7 CONCLUSIONS

The current HMF design pattern is extendible and modular. In our opinion the HMF succeeds in its purpose. We have confidence that HMF comes as a relief to those application domains that so far have been relatively underequipped with practical decision support tools, due to the lack of 'hard and solid' domain knowledge that can be used as a basis for probabilistic models.

The arrangement of the HMF supports a working method which deals with tunnel-view in a well considered manner. The HMF will not explicitly reduce or prevent bias occurring within the topology of a model. However, it offers the possibility to use certain strategies during the design of a model which lead to more balanced and thus less biased models. Using such strategies will enlarge the awareness about tunnel-view (and bias) and as such may partly prevent it.

Although the requirements of reliability and usability are not validated by domain experts and analysts, several issues concerning these requirements have been discussed in this paper. The Asia example shows that posteriors via a HMF model can be quite similar to those derived via a belief network based on causality. The issues that we have encountered so far in applying belief networks for criminal investigations have been addressed in this paper. However, it is a continuous effort to further improve the HMF.

8 FUTURE RESEARCH

One of the complementary wishes of the authors involves a bias measurement combined with automated commentary that highlights useful missing evidence. By calculating how discriminative the indicators and the evidence is to each hypothesis and counterhypothesis, we can evaluate whether tunnel vision may be present. It can also be used to investigate the added value of collecting evidence for unobserved indicators. One way of getting this information is by simulating evidence and evaluate the posteriors of all hypotheses. Since the maximum potential impact of an indicator may only occur at a certain combination of evidence for other indicators, the simulation should consider all possible combinations of evidence for all unobserved indicators. This may be a costly operation. Alternatively one may derive the maximum impact directly from the conditional probability tables of the variables, and use message passing to investigate the maximum potential impact of each indicator.

The naive structure of a HMF belief network may in some occasions not capture the targeted effects. In those cases we would like to extend the HMF model with constraining variables that model the synergistic effect between indicators (or in between hypotheses). We have not been able to test such mechanisms in realistic cases so far. Therefore these need further investigation to test the feasibility of adding constraints, and whether the implications of such mechanisms violate the extendibility and modularity.

The HMF has been applied on several study cases based on real data by the authors. In the foreseeable future it is expected that domain experts will work with this framework. Their experience will be very useful for validating the usability and reliability of this method, and for finding ways to further improve it.

References

[1] B. Abramson, J. Brown, W. Edwards, A. Murphy, and R.L. Winkler. Hailfinder: A bayesian system for forecasting severe weather. *International Journal of Forecasting*, 12(1):57–72, 1996.

[2] Ingo Beinlich, Jaap Suermondt, Martin Chavez, and Gregory Cooper. The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks. In *Second European Conference on Artificial Intelligence in Medicine*, 1988.

[3] A. Biedermann and F. Taroni. Bayesian networks and probabilistic reasoning about scientific evidence when there is a lack of data. *Forensic Science International*, (157):163–167, 2006.

[4] F. J. Díez and M. J. Druzdzel. Canonical probabilistic models for knowledge engineering. Technical Report CISIAD-06-01, UNED, Madrid, Spain, 2006.

[5] Marek J. Druzdzel and Linda C. van der Gaag. Elicitation of probabilities for belief networks: Combining qualitative and quantitative information. In *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, pages 141–148. Morgan Kaufmann, 1995.

[6] N Friedman, D Geiger, and M Goldszmidt. Bayesian network classifiers. In *Machine Learning*, volume 29, pages 131–163, 1997.

[7] Gerd Gigerenzer and Ulrich Hoffrage. How to improve bayesian reasoning without instruction: Frequency formats. *Psychological Review*, 102:684–704, October 1995.

[8] Max Henrion. Some practical issues in constructing belief networks. In *Proceedings of the 3rd Annual Conference on Uncertainty in Artificial Intelligence (UAI-87)*, pages 161–174, New York, NY, 1987. Elsevier Science.

[9] P.E.M. Huygen. Use of bayesian belief networks in legal reasoning. In *17th BILETA Annual Conference*, 2002.

[10] J. H. Kim and J. Pearl. A computation model for causal and diagnostic reasoning in inference systems. *Proceedings of the 8th International Joint Conference on AI*, pages 190–193, 1983.

[11] M. Korver and P. Lucas. Converting a rule-based expert system into a belief network. *Medical Informatics*, 18(3):219–241, 1993.

[12] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Readings in uncertain reasoning table of contents*, pages 415–448, 1990.

[13] J. Lin. Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151, 1991.

[14] Krol Kevin Mathias, Cynthia Isenhour, Alex Dekhtyar, Judy Goldsmith, and Beth Goldstein. When domains require modeling adaptations. In *4th Bayesian Modelling Applications Workshop at UAI*, 2006.

[15] V Metsis, I Androutsopoulos, and G Paliouras. Spam filtering with naive bayes: which naive bayes. In *In 3rd Conference on Email and Anti-Spam, Mountain View, ca*, 2006.

[16] Sucheta Nadkarni and Prakash P. Shenoy. A causal mapping approach to constructing bayesian networks. *Decision Support Systems*, 38(2):259–281, November 2004.

[17] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

[18] H. Prakken and S. Renooij. Reconstructing causal reasoning about evidence: a case study. In *Legal Knowledge and Information Systems. JURIX 2001: The Fourteenth Annual Conference*, pages 131–142. IOS Press, Amsterdam, The Netherlands, 2001.

[19] Rita Sharma, David Poole, and Clinton Smyth. A system for ontologically-grounded probabilistic matching. In *Proceedings of the Fifth UAI Bayesian Modeling Applications Workshop*, 2007.

[20] B.W. Wisse, S.P. van Gosliga, N.P. van Elst, and A.I. Barros. Relieving the elicitation burden of bayesian belief networks. *Sixth Bayesian Modelling Applications Workshop on UAI 2008.*

An Experimental Procedure for Evaluating User-Centered Methods for Rapid Bayesian Network Construction

Michael Farry, Jonathan Pfautz & Zach Cox
Charles River Analytics, Inc.
625 Mount Auburn St.
Cambridge, MA 02138

Ann Bisantz & Richard Stone
Industrial and Systems Engineering
University at Buffalo
Amherst NY 14260

Emilie Roth
Roth Cognitive Engineering
Brookline, MA

Abstract

Bayesian networks (BNs) are excellent tools for reasoning about uncertainty and capturing detailed domain knowledge. However, the complexity of BN structures can pose a challenge to domain experts without a background in artificial intelligence or probability when they construct or analyze BN models. Several canonical models have been developed to reduce the complexity of BN structures, but there is little research on the accessibility and usability of these canonical models, their associated user interfaces, and the contents of the models, including their probabilistic relationships. In this paper, we present an experimental procedure to evaluate our novel Causal Influence Model structure by measuring users' ability to construct new models from scratch, and their ability to comprehend previously constructed models. [Results of our experiment will be presented at the workshop.]

1. INTRODUCTION AND MOTIVATION

A Bayesian network (BN) (Jensen, 2001; Pearl, 1988) is a probabilistic model used to reason under uncertainty. Successful efforts in applying Bayesian modeling to a variety of domains (e.g., computer vision (Rimey & Brown, 1994), social networks (Koelle et al., 2006), human cognition (Guarino et al., 2006; Glymour, 2001), and disease detection (Pang et al., 2004)) have inspired knowledge engineers to use BNs to capture domain knowledge from experts. However, expressing an expert's domain knowledge in a BN is cumbersome due to the complex, tedious, and mathematical nature of conditional probability table (CPT) construction. Adding states and parents to a node quickly results in an exponential explosion in the number of CPT entries required (Pfautz et al., 2007). Canonical models such as Noisy-OR (Henrion, 1989; Pearl, 1988), Noisy-MAX (Diez & Galan, 2003; Diez, 1993; Henrion, 1989), Qualitative Probabilistic Networks (Wellman, 1990) and Influence Networks (IN) (Jensen, 1996; Rosen & Smith, 1996a; Rosen & Smith, 1996b) have been developed to mitigate this problem. In response to some issues raised by those models, and to simplify the Bayesian modeling process through novel user interface techniques, we developed a new canonical model, the Causal Influence Model (CIM) (Cox & Pfautz, 2007; Pfautz et al., 2007). The CIM paradigm was inspired by anecdotal evidence gained by developing systems for domain experts interacting with BNs and by an analysis of other canonical models to determine the constraints that limit their generalizability and applicability.

There have been few user-centered evaluation efforts to assess how (and if) canonical models help domain experts elicit their knowledge and understanding of models presented to them, or how graphical interfaces and their features and properties impact the way people create, interpret, reason with, or base actions on Bayesian networks. The purpose of our study is to provide baseline information on how people construct and describe CIMs presented and created within a graphical user interface.

1.1 BACKGROUND

A *canonical model* (Diez & Druzdzel, 2001) is a modeling pattern that allows probabilistic relationships between nodes to be specified by a reduced set of parameters (i.e., without completing every cell in a CPT). By assuming that the reduced parameters can still accurately represent the domain being modeled, users can quickly build a complex BN that would otherwise take a large amount of time. Most canonical models achieve their reduced parameters by assuming the independent effects of parents. This assumption allows a linear number of parameters to quantify an entire CPT; in the best-case scenario, only a single parameter per parent is needed. Canonical models can also serve as a "front-end" tool for the initial model-building effort, since the CPTs can always be refined by hand or with data at a later time. Some of the simplified patterns followed by canonical models have been motivated by the process followed when eliciting key factors and probabilistic relationships

from domain experts (O'Hagan et al., 2006; Hastie & Dawes, 2001).

A review of canonical models sheds light on the advantages and drawbacks of each model. The Influence Network (IN) model can only be used with Boolean nodes. It assumes that the child node has a baseline probability of occurring independently of any parent effects and that each parent independently influences the child to be more or less likely to be true. Since a single baseline probability for the child and a single change in probability for each parent are simple parameters for users to specify, the IN represents a powerful mechanism for capturing domain knowledge. However, since only Boolean nodes are allowed in the IN model, model flexibility is significantly reduced. BNs commonly contain nodes that represent concepts other than the occurrence or non-occurrence of events, and INs cannot be used to simplify these BNs without considerably re-architecting the model.

The Noisy-OR model is also used only with Boolean nodes and assumes that a true state in any parent can cause the child to be true independently of the other parents, with some uncertainty. Similar to INs, the main drawback of the Noisy-OR is its limitation to only Boolean nodes. The Noisy-MAX model generalizes the Noisy-OR and allows ordinal nodes at the expense of increasing the complexity of parameters. Although Noisy-MAX does work with ordinal nodes, it cannot be used with more general discrete nodes that do not have ordered states. These nodes, referred to as categorical nodes, have an arbitrary number of unordered states and usually represent the category or type of something. Qualitative Probabilistic Networks (QPNs) allow for the construction of purely qualitative relationships between nodes in a network, to abstract from the highly quantitative and numerical nature of typical Bayesian models. QPNs consider the "signs" inherent in probabilistic relationships between nodes, and consider the additive synergies between nodes to capture more complicated probabilistic relationships between them (i.e., if A and B both have a positive influence on node C, their influences may be synergistic in nature: if A and B are both true, their cumulative influence upon C may be greater than just the sum of their individual influences.) QPNs allow for more qualitative model elicitation and may therefore be appropriate for interactions with non-technical experts, but they are limited in their ability to provide hard, numerical estimates of the likelihood of events.

The Causal Influence Model (CIM) is a canonical model that retains the desirable properties of the IN while providing solutions to its problems. The CIM assumes that each node is discrete and has an arbitrary number of states with arbitrary meaning. Each node has a baseline probability distribution, independent of any parent effects. Each parent independently influences these baseline probabilities to be more or less likely. The CIM also introduces simplifications that govern the generation of conditional probability relationships, enabling Boolean, ordinal, and categorical nodes to be included. A full description of the mathematical formulas that govern CIMs, including formulas to translate CIM link strengths into conditional probability tables, is provided in (Cox et al., 2007).

Studies have been conducted to analyze and mitigate complexities that arise in the construction of Bayesian models as a result of knowledge elicitation (Onisko, Druzdzel, & Wasyluk, 2001), but no studies to date have assessed the accessibility and usability of various canonical models and associated user interfaces when provided directly to domain experts. The following study investigates how users interpret and create CIMs within a particular user interface.

2. METHOD

2.1 PARTICIPANTS

Up to twenty participants are recruited from the university community to perform the study. After providing informed consent, participants are given the Ishihara Test for color blindness. Participants who pass this screening continue with the study.

2.2 EXPERIMENTAL SYSTEM

We have developed an CIM-enabled version of our BNet.Builder product to allow us to experiment with graphical interfaces for Bayesian network modeling (Pfautz et al., 2007). Using a simple point-and-click interface, users can create, label, connect, and move nodes in the model. Users can also create and modify causal links to represent positive or negative influences between nodes and the strength of those relationships. Users can also post or remove evidence to any node and view the effects of posted evidence on the belief states of other nodes. Link strengths are converted using CPTs based on algorithms provided in (Cox et al., 2007; Pfautz et al., 2007). The positivity or negativity of a causal link and the link strength are represented visually by the color and thickness of the link, respectively.

To simplify model construction for this particular experiment, the CIM interface has been constrained so that all nodes are Boolean; initial beliefs are set to 0.5 for all nodes and cannot be changed directly by the user (but can change based on evidence or link strengths); and only "hard" evidence can be posted (e.g., evidence that the node was either fully true, or fully false). This represents a set of simplifications we have found useful in other work, particularly among users less familiar with Bayesian modeling techniques. Our main goal in this study is to determine whether participants can reason about previously constructed CIMs and construct models to match a given situation. Since these are specific, novel, and fundamental questions with little previous research behind them, we have started with a simple case. The

inclusion of additional node types, in particular, is useful for future work in comparing CIMs to other canonical models such as INs, Noisy-OR, and Noisy-MAX.

2.3 EXPERIMENTAL TASKS

Participants will be asked to provide descriptions of and answer questions about a series of CIMs shown in the BNet.Builder interface. In the first task, participants will be shown a model and asked questions about the structure and nature of relationships in the model (specifically, questions asking them to describe elements of the model, and questions related to abductive and deductive reasoning using the model). For instance, given the following example model (Figure 1), participants would be asked:

- *Description:* This picture shows a model of part of a car. Describe what causes headlights to be dim, or not dim.
- *Abductive Reasoning:* If the headlights are dim, what does that mean about the other parts of the car?
- *Deductive Reasoning:* The alternator is working. What does that suggest about the headlights? The battery is old. What does that suggest about the headlights? What if the battery is new and the alternator is failing?

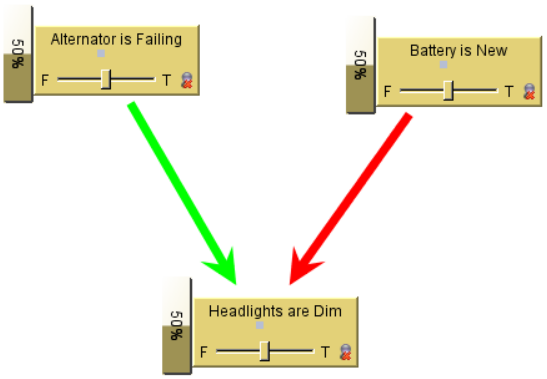


Figure 1. Example model used in the experiment. The green link represents positive influence, while the red link represents negative influence within our CIM-enabled interface.

In the second task, participants can manipulate the causal links and post evidence to see how changing the strength and directionality of the links between the nodes, and evidence about the state of the nodes, affects beliefs about whether the nodes are true or false. They will respond to similar sets of questions as provided in the first task. Finally, in the third task, participants will be asked to construct models from scratch using the interface based on several different vignettes, such as the following:

> *The headlight system on a car is dependent on two components: a battery, which stores energy to power the lights, and an alternator, which converts mechanical energy from the car's engine into stored energy in the battery. When the car is running, the alternator "recharges" the battery. This process only works if the alternator is working, and the battery is new.*

Four models/vignettes have been constructed for each task (a total of 12). Each model has the following relationships: 1 child/1 parent, 2 children/1 parent, 1 child/2 parents, 2 children/2 parents. In all cases, all children are linked to all parents. Also, in all but the 1 child/1 parent case, one parent-child link is negative. This simplification provides the basis for the initial study. We expect to expand upon this simple representation with later empirical work.

2.4 INDEPENDENT VARIABLE

Two stimuli sets are created based on the 12 models. Either the nodes in the models (or phrases in the vignette) are phrased positively, or they include at least one node that uses negative phrasing (e.g., "battery is not new"). This difference allows us to investigate how semantic properties of the model or situation affect task performance. This condition has been inspired by our experience in domain expert interaction with CIM modeling interfaces, where we observed the articulation of variable names as a source of common confusion. The use of negatives in the variable name (e.g., "not raining") or logical antonyms (e.g., "happy" and "sad") tends to lead to later confusion in expressing causal relationships (e.g., "if it is not not-raining, then it is unlikely that Rakesh will not bring his umbrella"). By including this specific independent variable, we will be able to assess which specific patterns of reasoning are most difficult for users. Participants are randomly assigned to one of the two stimuli sets (up to 10 participants per condition). This sample size is consistent with those used in usability type tests, and will allow us to analyze verbal protocols of participants to look for patterns across conditions.

2.5 DEPENDENT MEASURES AND ANALYSIS

Throughout all three tasks, participants are asked to "talk aloud" while performing the task to describe how they are thinking about or creating the models. Screen capture software is used to record participants' interaction with and construction of models. Participants are also fitted with a view point eye tracker (lightweight glasses that have an attached camera that tracks the corneal movements of the participant's eye to assess gaze relative to the computer screen they are working on). The eye tracking system is used to record aspects of gaze position and dwell time at a screen location. Time to complete the tasks is also being recorded.

Data from the audio, eye track, and screen capture processes is combined to create a "process trace" of each participant's behavior describing and creating CIMs (Woods, 1993). Verbalizations and actions are coded and analyzed (Bainbridge & Sanderson, 1995; Sanderson & Fisher, 1994; Woods, 1993) to identify the correctness

and completeness of the descriptions and answers provided by participants in the first task, the processes with which participants constructed the models in the second task, and the form and content of the models produced in the third task.

3. ANTICIPATED RESULTS AND DISCUSSION

The purpose of this study is to provide baseline information regarding how people construct and describe CIM models presented and created within the BNet.Builder interface. There is continued interest in simplifying the manner in which domain expertise is elicited, and the creation and presentation of Bayesian network models through direct manipulation and visualization. However, information on how these tools are used by practitioners, how they affect the models that people produce, and how they affect the way that people interpret models or predict outcomes is missing. We anticipate that users will have more difficulty explaining and constructing models with more parent-child connections. We also anticipate users having more difficulty explaining and constructing models when there are more nodes with negative causal links because of the increase in complexity of the models.

In this study, we intend to measure reasoning patterns involving negative quantities that give users the most trouble. We anticipate that users will have the most difficulty interpreting and creating models when nodes are presented with "negatively phrased" labels (e.g., assessing the influence of a node labeled "battery is not new" on a node labeled "headlights are dim"). If this is the case, it suggests a need for developers of CIMs (and BNs in general) to encourage users to employ certain modeling patterns, possibly by constraining the description of nodes. These constraints, in turn, can be accomplished through prior training or interface wizards, or through intelligent, automatic processing of user entries, and provision of suggested alternatives (e.g., pop-up suggestions). These interventions could be tested in further studies.

The primary contribution of this paper will be process- and product-oriented descriptions of how this graphical tool is used to interpret and create CIMs. Future research could compare how models created within the CIM framework compare to those using more traditional BN structures, from the point of view of the user. This study used simple Bayesian models, with constrained parameters and interaction capabilities, and used only Boolean nodes. Future studies, guided by these initial findings, can be conducted using more complex models, a greater variety of node types (e.g., categorical, ordinal), and allow subjects greater flexibility in manipulating CPTs and posting evidence. Other issues for investigation include measuring and mitigating user tendencies to confuse "evidence" and "belief" (both as terms, and in the values these terms represent), measuring tendencies to disregard parental independence when constructing CIMs, and further observation of user reaction to non-intuitive but correct behavior (e.g., becoming confused when particular variables appear overly sensitive or insensitive to posted evidence.)

The CIM interface provides a user-friendly way to express causal influences between nodes, vastly decreasing the number of parameters needed to construct causal models and providing the capability for a much broader base of users to perform Bayesian modeling. Within the experimental interface, participants express relative degrees of influence over a range of 11 steps (from positive to negative 5, with a neutral intermediate value). Additional studies are necessary to clarify the appropriate level of granularity of influence assignment (e.g., 3 steps? 11 steps? 51 steps?) as well whether other methods of assigning strengths across sets of links (e.g., normalized strengths, rank ordered strengths) have merit. Finally, detailed studies with real-world models, situations, and domain experts are required.

Acknowledgements

We would like to thank David Koelle, Geoffrey Catto, Joseph Campolongo, Sam Mahoney, Sean Guarino, and Eric Carlson for their contributions in the development of the CIM and identifying hypotheses to investigate. We also extend our deepest gratitude to Greg Zacharias for his continued funding and support of our work with Bayesian networks.

References

Bainbridge, L., & Sanderson, P. (2005). Verbal protocol analysis. In J. R. Wilson & E. N. Corlett (Eds.), Evaluation of Human Work (pp. 159 - 184). Boca Raton: Taylor and Francis.

Cox, Z. & Pfautz, J. (2007). *Causal Influence Models: A Method for Simplifying Construction of Bayesian Networks.* (Rep. No. R-BN07-01). Cambridge, MA: Charles River Analytics Inc.

Diez, F. J. (1993). Parameter Adjustment in Bayes Networks: The Generalized Noisy OR-Gate. In *Proceedings of the 9th Conference of Uncertainty in Artificial Intelligence,* (pp. 99-105). San Mateo, CA: Morgan Kaufmann.

Diez, F. J. & Druzdzel, M. J. (2001). Fundamentals of Canonical Models. In *Proceedings of Ponencia Congreso: IX Conferencia De La Asociacion Espanola Para La Inteligencia Artificial (CAEPIA-TTIA 2001),* (pp. 1125-1134).

Diez, F. J. & Galan, S. F. (2003). An Efficient Factorization for the Noisy MAX. *International Journal o Intelligent Systems,* 18165-177.

Glymour, C. (2001). *The Mind's Arrows: Bayes Nets and Graphical Causal Models in Psychology*. Cambridge, MA: The MIT Press.

Guarino, S., Pfautz, J., Cox, Z., & Roth, E. (2006). Modeling Human Reasoning About Meta-Information. In *Proceedings of 4th Bayesian Modeling Applications Workshop at the 22nd Annual Conference on Uncertainty in AI: UAI '06*. Cambridge, Massachusetts.

Hastie, R. & Dawes, R. M. (2001). *Rational Choice in an Uncertain World: The Psychology of Judgment and Decision-Making*. London, UK: Sage Publications.

Henrion, M. (1989). Some Practical Issues in Constructing Belief Networks. In L. Kanal, T. Levitt, & J. Lemmer (Eds.), *Uncertainty in Artificial Intelligence 3* (pp. 161-173). North Holland: Elsevier Science Publishers.

Jensen, F. V. (1996). *An Introduction to Bayesian Networks*. London: University College London Press.

Jensen, F. V. (2001). *Bayesian Networks and Decision Graphs*. New York: Springer-Verlag.

Koelle, D., Pfautz, J., Farry, M., Cox, Z., Catto, G., & Campolongo, J. (2006). Applications of Bayesian Belief Networks in Social Network Analysis. In *Proceedings of 4th Bayesian Modeling Applications Workshop at the 22nd Annual Conference on Uncertainty in AI: UAI '06*. Cambridge, Massachusetts.

Kraaijeveld, P., Druzdzel, M., Onisko, A., & Wasyluk, H. (2005). GeNIeRate: An Interactive Generator of Diagnostic Bayesian Network Models. In *Proceedings of Working Notes of the 16th International Workshop on Principles of Diagnosis (DX-05),* (pp. 175-180).

O'Hagan, A., Buck, C., Daneshkhah, A., Eiser, R., Garthwaite, P., Jenkinson, D. et al. (2006). *Uncertain Judgements: Eliciting Experts' Probabilities*. New York: Wiley & Sons.

Onisko, A., Druzdzel, M., & Wasyluk, H. (2001). Learning Bayesian Network Parameters From Small Data Sets: Application of Noisy-OR Gates. *International Journal of Approximate Reasoning,* 27(2), 165-182.

Pang, B., Zhang, D., Li, N., & Wang, K. (2004). Computerized Tongue Diagnosis Based on Bayesian Networks. *IEEE Transactions on Biomedical Engineering,* 51(10), 1803-1810.

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

Pfautz, J., Cox, Z., Koelle, D., Catto, G., Campolongo, J., & Roth, E. (2007). User-Centered Methods for Rapid Creation and Validation of Bayesian Networks. In *Proceedings of 5th Bayesian Applications Workshop at Uncertainty in Artificial Intelligence (UAI '07)*. Vancouver, British Columbia.

Rimey, R. & Brown, C. (1994). Control of Selective Perception Using Bayes Nets and Decision Theory. *International Journal of Computer Vision,* 12(2-3), 173-207.

Rosen, J. & Smith, W. (1996a). Influence Net Modeling With Causal Strengths: An Evolutionary Approach. In *Proceedings of Command and Control Research and Technology Symposium*.

Rosen, J. A. & Smith, W. L. (1996b). Influencing Global Situations: A Collaborative Approach. *US Air Force Air Chronicles*.

Sanderson, P. M., & Fisher, C. (1994). Exploratory sequential data analysis. *Human Computer Interaction*, 9(3), 251 - 317.

Van der Gagg, L.C., Geenen, P.L., & Tabachneck-Schijf, H.J.M. (2006). Verifying Monotonicity of Bayesian Networks with Domain Experts. In *Proceedings of 4th Bayesian Modeling Applications Workshop at the 22nd Annual Conference on Uncertainty in AI: UAI '06*. Cambridge, Massachusetts.

Woods, D. D. (1993). Process tracing methods for the study of cognition outside of the experimental psychology laboratory. In G. A. Klein, J. Orasanu, R. Calderwood & C. E. Zsambok (Eds.), *Decision-making in action: Models and Methods* (pp. 228 - 251). Norwood NJ: Ablex Publishers.

Wellman, M. P. (1990). Fundamental Concepts of Qualitative Probabilistic Networks. *Artificial Intelligence,* 44(3), 257-303.

The Impact of Overconfidence Bias on Practical Accuracy of Bayesian Network Models: An Empirical Study

Marek J. Drużdżel[1,2] **& Agnieszka Oniśko**[1,3]
[1] Faculty of Computer Science, Białystok Technical University, Wiejska 45A, 15-351 Białystok, Poland
[2] Decision Systems Laboratory, School of Information Sciences and Intelligent Systems Program, University of Pittsburgh, Pittsburgh, PA 15260, USA
[3] Magee Womens Hospital, University of Pittsburgh Medical Center, Pittsburgh, PA 15260, USA

Abstract

In this paper, we examine the influence of overconfidence in parameter specification on the performance of a Bayesian network model in the context of Hepar II, a sizeable Bayesian network model for diagnosis of liver disorders. We enter noise in the parameters in such a way that the resulting distributions become biased toward extreme probabilities. We believe that this offers a systematic way of modeling expert overconfidence in probability estimates. It appears that the diagnostic accuracy of Hepar II is less sensitive to overconfidence in probabilities than it is to underconfidence and to random noise, especially when noise is very large.

1 INTRODUCTION

Decision-analytic methods provide an orderly and coherent framework for modeling and solving decision problems in decision support systems [5]. A popular modeling tool for complex uncertain domains is a Bayesian network [13], an acyclic directed graph quantified by numerical parameters and modeling the structure of a domain and the joint probability distribution over its variables. There exist algorithms for reasoning in Bayesian networks that typically compute the posterior probability distribution over some variables of interest given a set of observations. As these algorithms are mathematically correct, the ultimate quality of reasoning depends directly on the quality of the underlying models and their parameters. These parameters are rarely precise, as they are often based on subjective estimates. Even when they are based on data, they may not be directly applicable to the decision model at hand and be fully trustworthy.

Search for those parameters whose values are critical for the overall quality of decisions is known as sensitivity analysis. Sensitivity analysis studies how much a model output changes as various model parameters vary through the range of their plausible values. It allows to get insight into the nature of the problem and its formalization, helps in refining the model so that it is simple and elegant (containing only those factors that matter), and checks the need for precision in refining the numbers [8]. It is theoretically possible that small variations in a numerical parameter cause large variations in the posterior probability of interest. Van der Gaag and Renooij [17] found that practical networks may indeed contain such parameters. Because practical networks are often constructed with only rough estimates of probabilities, a question of practical importance is whether overall imprecision in network parameters is important. If not, the effort that goes into polishing network parameters might not be justified, unless it focuses on their small subset that is shown to be critical.

There is a popular belief, supported by some anecdotal evidence, that Bayesian network models are overall quite tolerant to imprecision in their numerical parameters. Pradhan et al. [14] tested this on a large medical diagnostic model, the CPCS network [7, 16]. Their key experiment focused on systematic introduction of noise in the original parameters (assumed to be the gold standard) and measuring the influence of the magnitude of this noise on the average posterior probability of the true diagnosis. They observed that this average was fairly insensitive to even very large noise. This experiment, while ingenious and thought provoking, had two weaknesses. The first of these, pointed out by Coupé and van der Gaag [3], is that the experiment focused on the average posterior rather than individual posterior in each diagnostic case and how it varies with noise, which is of most interest. The second weakness is that the posterior of the correct diagnosis is by itself not a sufficient measure of model robustness. The weaknesses of this experiment were also discussed in [6] and [9]. In our earlier work [9], we replicated the experiment of Pradhan et al. using

HEPAR II, a sizeable Bayesian network model for diagnosis of liver disorders. We systematically introduced noise in HEPAR II's probabilities and tested the diagnostic accuracy of the resulting model. Similarly to Pradhan et al., we assumed that the original set of parameters and the model's performance are ideal. Noise in the original parameters led to deterioration in performance. The main result of our analysis was that noise in numerical parameters started taking its toll almost from the very beginning and not, as suggested by Pradhan et al., only when it was very large. The region of tolerance to noise, while noticeable, was rather small. That study suggested that Bayesian networks may be more sensitive to the quality of their numerical parameters than popularly believed. Another study that we conducted more recently [4] focused on the influence of progressive rounding of probabilities on model accuracy. Here also, rounding had an effect on the performance of HEPAR II, although the main source of performance loss were zero probabilities. When zeros introduced by rounding are replaced by very small non-zero values, imprecision resulting from rounding has minimal impact on HEPAR II's performance.

Empirical studies conducted so far that focused on the impact of noise in probabilities on Bayesian network results disagree in their conclusions. Also, the noise introduced in parameters was usually assumed to be random, which may not be a reasonable assumption. Human experts, for example, often tend to be overconfident [8]. This paper describes a follow-up study that probes the issue of sensitivity of model accuracy to noise in probabilities further. We examine whether a bias in the noise that is introduced into the network makes a difference. We enter noise in the parameters in such a way that the resulting distributions become biased toward extreme probabilities. We believe that this offers a systematic way of modeling expert overconfidence in probability estimates. Our results show again that the diagnostic accuracy of HEPAR II is sensitive to imprecision in probabilities. It appears, however, that the diagnostic accuracy of HEPAR II is less sensitive to overconfidence in probabilities than it is to random noise. We also test the sensitivity of HEPAR II to underconfidence in parameters and show that underconfidence in paramaters leads to more error than random noise.

The remainder of this paper is structured as follows. Section 2 introduces the HEPAR II model. Section 3 describes how we introduced noise into our probabilities. Section 4 describes the results of our experiments. Finally, Section 5 discusses our results in light of previous work.

2 THE HEPAR II MODEL

Our experiments are based on HEPAR II [10, 11], a Bayesian network model consisting of over 70 variables modeling the problem of diagnosis of liver disorders. The model covers 11 different liver diseases and 61 medical findings, such as patient self-reported data, signs, symptoms, and laboratory tests results. The structure of the model, (i.e., the nodes of the graph along with arcs among them) was built based on medical literature and conversations with domain experts and it consists of 121 arcs. HEPAR II is a real model and it consists of nodes that are a mixture of propositional, graded, and general variables. There are on the average 1.73 parents per node and 2.24 states per variable. The numerical parameters of the model (there are 2,139 of these in the most recent version), i.e., the prior and conditional probability distributions, were learned from a database of 699 real patient cases. Readers interested in the HEPAR II model can download it from Decision Systems Laboratory's model repository at `http://genie.sis.pitt.edu/`.

As our experiments study the influence of precision of HEPAR II's numerical parameters on its accuracy, we owe the reader an explanation of the metric that we used to test the latter. We focused on diagnostic accuracy, which we defined in our earlier publications as the percentage of correct diagnoses on real patient cases. When testing the diagnostic accuracy of HEPAR II, we were interested in both (1) whether the most probable diagnosis indicated by the model is indeed the correct diagnosis, and (2) whether the set of w most probable diagnoses contains the correct diagnosis for small values of w (we chose a "window" of w=1, 2, 3, and 4). The latter focus is of interest in diagnostic settings, where a decision support system only suggest possible diagnoses to a physician. The physician, who is the ultimate decision maker, may want to see several alternative diagnoses before focusing on one.

With diagnostic accuracy defined as above, the most recent version of the HEPAR II model reached the diagnostic accuracy of 57%, 69%, 75%, and 79% for window sizes of 1, 2, 3, and 4 respectively [12].

3 INTRODUCTION OF NOISE INTO HEPAR II PARAMETERS

When introducing noise into parameters, we used essentially the same approach as Pradhan et al. [14], which is transforming each original probability into log-odds function, adding noise parametrized by a parameter σ (as we will show, even though σ is proportional to the amount of noise, in our case it cannot be directly interpreted as standard deviation), and trans-

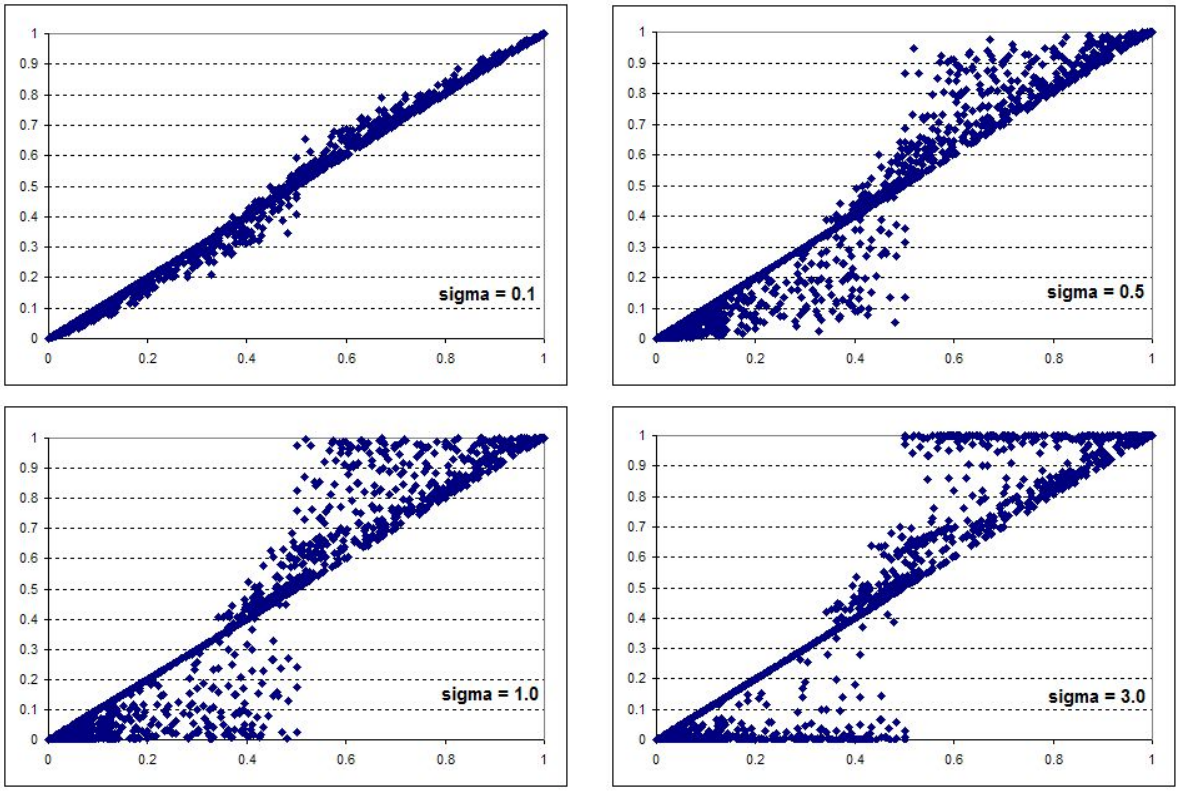


Figure 1: Transformed (biased, overconfident) vs. original probabilities for various levels of σ.

forming it back to probability, i.e.,

$$p' = Lo^{-1}[Lo(p) + \text{Noise}(0, \sigma)] \ , \tag{1}$$

where

$$Lo(p) = \log_{10}[p/(1-p)] \ . \tag{2}$$

3.1 Overconfidence bias

Now, we designed the Noise() function as follows. Given a discrete probability distribution Pr, we identify the smallest probability p_S. We transform this smallest probability p_S into p'_S by making it even smaller, according to the following formula:

$$p'_S = Lo^{-1}[Lo(p_S) - |\text{Normal}(0, \sigma)|] \ .$$

We make the largest probability in the probability distribution Pr, p_L larger by precisely the amount by which we decreased p_S, i.e.,

$$p'_L = p_L + p_S - p'_S \ .$$

We are by this guaranteed that the transformed parameters of the probability distribution Pr$'$ add up to 1.0.

Figure 1 shows the effect of introducing the noise. As we can see, the transformation is such that small probabilities are likely to become smaller and large probabilities are likely to become larger. Please note that distributions have become more biased towards the extreme probabilities. It is straightforward to prove that the entropy of Pr$'$ is smaller than the entropy of Pr. The transformed probability distributions reflect overconfidence bias, common among human experts.

An alternative way of introducing biased noise, suggested by one of the reviewers, is by means of building a logistic regression/IRT model (e.g., [1, 2, 15])for each conditional probability table and, subsequently, manipulating the slope parameter.

3.2 Underconfidence bias

Now, we designed the Noise() function as follows. Given a discrete probability distribution Pr, we identify the highest probability p_S. We transform this largest probability p_L into p'_L by making it smaller, according to the following formula:

$$p'_L = Lo^{-1}[Lo(p_L) - |\text{Normal}(0, \sigma)|] \ .$$

We make the smallest probability in the probability distribution Pr, p_S larger by precisely the amount by which we decreased p_L, i.e.,

$$p'_S = p_S + p_L - p'_L \ .$$

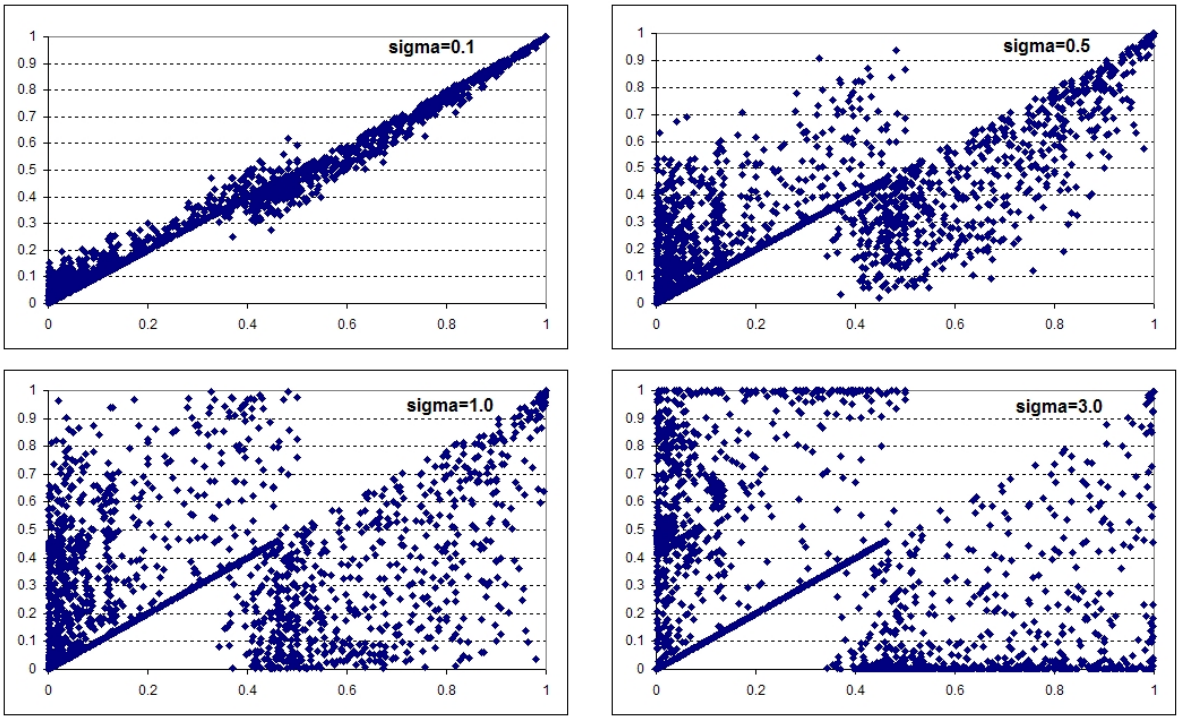


Figure 2: Transformed (biased, underconfident) vs. original probabilities for various levels of σ.

We are by this guaranteed that the transformed parameters of the probability distribution $\Pr'$ add up to 1.0.

Figure 2 shows the effect of introducing this noise. The transformed probability distributions reflect underconfidence bias.

3.3 Random noise

For illustration purpose, Figure 3 shows the transformation applied in our previous study [9]. For $\sigma > 1$ the amount of noise becomes so large that any value of probability can be transformed in any other value. This suggests strongly that $\sigma > 1$ is not really a region that is of interest in practice. The main reason why we look at such high σ values is that this was the range used in Pradhan et al. paper.

4 EXPERIMENTAL RESULTS

We have performed an experiment investigating the influence of biased noise in HEPAR II's probabilities on its diagnostic performance. For the purpose of our experiment, we assumed that the model parameters were perfectly accurate and, effectively, the diagnostic performance achieved was the best possible. Of course, in reality the parameters of the model may not be accurate and the performance of the model can be improved upon. In the experiment, we studied how this baseline performance degrades under the condition of noise, as described in Section 3.

We tested 30 versions of the network (each for a different standard deviation of the noise $\sigma \in < 0.0, 3.0 >$ with 0.1 increments) on all records of the HEPAR data set and computed HEPAR II's diagnostic accuracy. We plotted this accuracy in Figures 4 and 5 as a function of σ for different values of window size w.

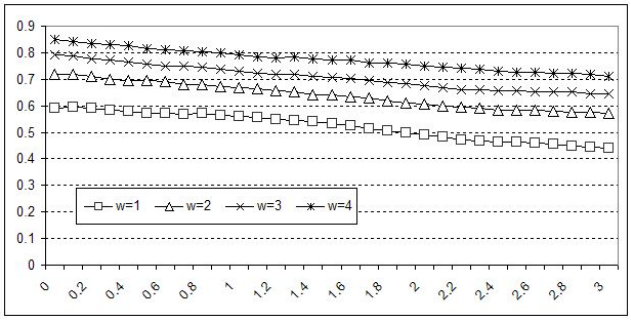


Figure 4: The diagnostic accuracy of HEPAR II for various window sizes as a function of the amount of biased overconfident noise (expressed by σ)
.

It is clear that HEPAR II's diagnostic performance deteriorates with noise. In order to facilitate comparison between biased and unbiased noise and, by this,

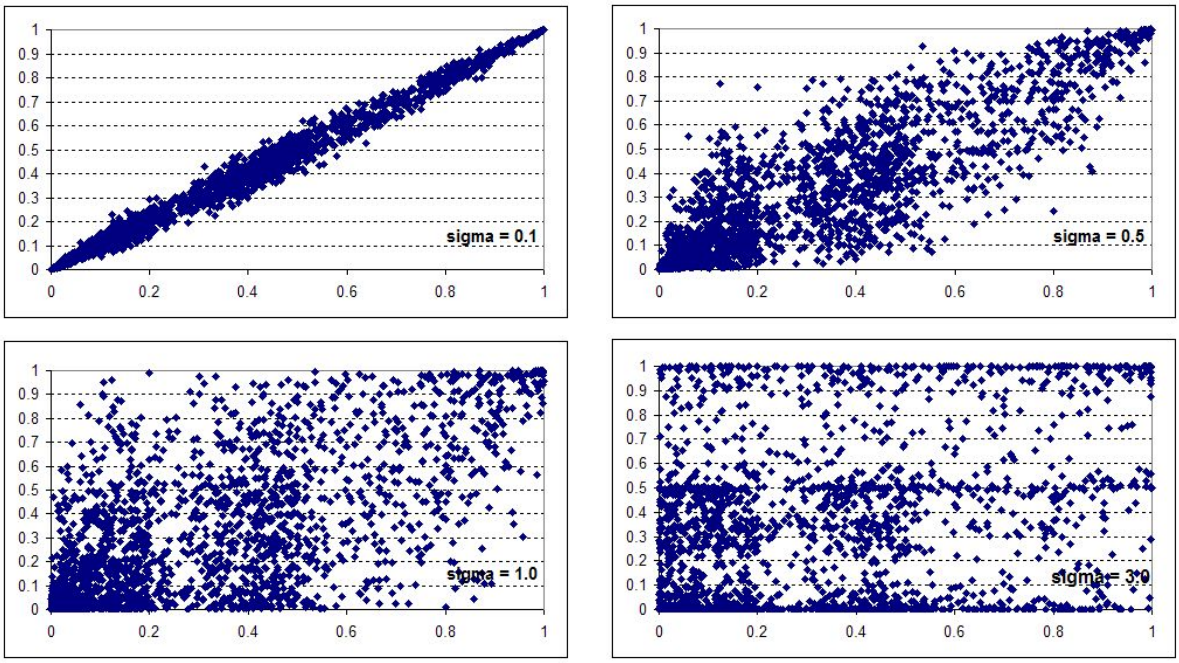


Figure 3: Transformed (unbiased) vs. original probabilities for various levels of σ.

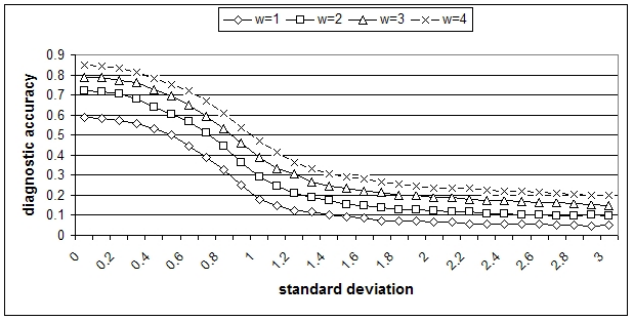


Figure 5: The diagnostic accuracy of HEPAR II for various window sizes as a function of the amount of biased underconfident noise (expressed by σ)
.

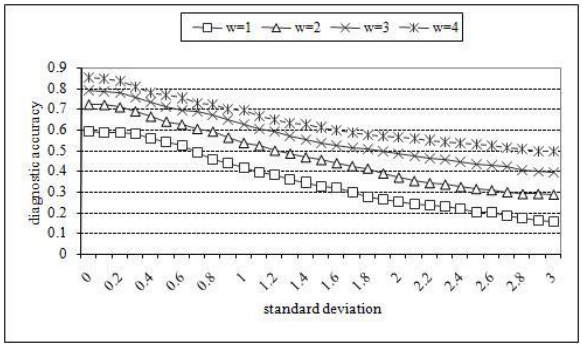


Figure 6: The diagnostic accuracy of HEPAR II for various window sizes as a function of amount of unbiased noise (expressed by σ) [9].

judgment of the influence of overconfidence bias on the results, we reproduce the experimental result of [9] in Figure 6. The results are qualitatively similar, although it can be seen that performance under overconfidence bias degrades more slowly with the amount of noise than performance under random noise. Performance under underconfidence bias degrades the fastest of the three. Figure 7 shows the accuracy of HEPAR II ($w = 1$) for biased and unbiased noise on the same plot, where this effect is easier to see.

It is interesting to note that for small values of σ, such as $\sigma < 0.2$, there is only a minimal effect of noise on performance. This observation may offer some justification to the belief that Bayesian networks are not too sensitive to imprecision of their probability parameters.

5 SUMMARY

This paper has studied the influence of bias in parameters on model performance in the context of a practical medical diagnostic model, HEPAR II. We believe that the study was realistic in the sense of focusing on a real, context-dependent performance measure. Our study has shown that the performance of HEPAR II is sensitive to noise in numerical parameters, i.e., the diagnostic accuracy of the model decreases after introducing noise into numerical parameters of the model.

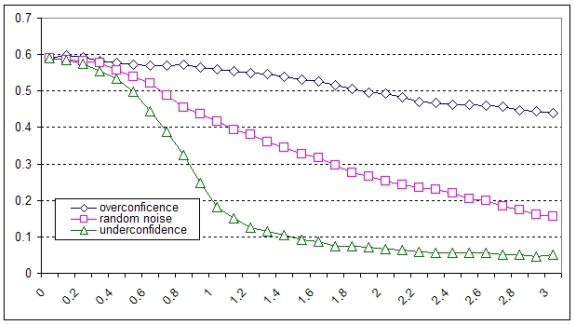


Figure 7: The diagnostic accuracy of HEPAR II as a function of the amount of noise (random, underconfident, and overconfident), window $w = 1$

While our result is merely a single data point that sheds light on the hypothesis in question, it looks like overconfidence bias has a smaller negative effect on model performance than random noise. Underconfidence bias leads to most serious deterioration of performance. While it is only a wild speculation that begs for further investigation, one might see our results as an explanation of the fact that humans tend to be overconfident rather than underconfident in their probability estimates.

Acknowledgments

This work was supported by the Air Force Office of Scientific Research grant FA9550-06-1-0243, by Intel Research, and by the MNiI (Ministerstwo Nauki i Informatyzacji) grant 3T10C03529. We thank Linda van der Gaag for suggesting extending our earlier work on sensitivity of Bayesian networks to precision of their numerical parameters by introducing bias in the noise. Reviewers for The Sixth Bayesian Modelling Applications Workshop provided several useful suggestions that have improved the readability and extended the scope of the paper.

The HEPAR II model was created and tested using SMILE, an inference engine, and GeNIe, a development environment for reasoning in graphical probabilistic models, both developed at the Decision Systems Laboratory, University of Pittsburgh, and available at `http://genie.sis.pitt.edu/`. We used SMILE in our experiments and the data pre-processing module of GeNIe for plotting scatter plot graphs in Figure 1.

References

[1] Russell G. Almond, Louis V. Dibello, F. Jenkins, R.J. Mislevy, D. Senturk, L.S. Steinberg, and D. Yan. Models for conditional probability tables in educational assessment. In T. Jaakkola and T. Richardson, editors, *Artificial Intelligence and Statistics 2001*, pages 137–143. Morgan Kaufmann, 2001.

[2] Russell G. Almond, Louis V. DiBello, Brad Moulder, and Juan-Diego Zapata-Rivera. Modeling diagnostic assessment with Bayesian networks. *Journal of Educational Measurement*, 44(4):341–359, 2007.

[3] Veerle H. M. Coupé and Linda C. van der Gaag. Properties of sensitivity analysis of Bayesian belief networks. *Annals of Mathematics and Artificial Intelligence*, 36:323–356, 2002.

[4] Marek J. Druzdzel and Agnieszka Oniśko. Are Bayesian networks sensitive to precision of their parameters? In S.T. Wierzchoń M. Kłopotek, M. Michalewicz, editor, *Intelligent Information Systems XVI, Proceedings of the International IIS'08 Conference*, pages 35–44, Warsaw, Poland, 2008. Academic Publishing House EXIT.

[5] Max Henrion, John S. Breese, and Eric J. Horvitz. Decision Analysis and Expert Systems. *AI Magazine*, 12(4):64–91, Winter 1991.

[6] O. Kipersztok and H. Wang. Another look at sensitivity analisis of Bayesian networks to imprecise probabilities. In *Proceedings of the Eight International Workshop on Artificial Intelligence and Statistics (AISTAT-2001)*, pages 226–232, San Francisco, CA, 2001. Morgan Kaufmann Publishers.

[7] B. Middleton, M.A. Shwe, D.E. Heckerman, M. Henrion, E.J. Horvitz, H.P. Lehmann, and G.F. Cooper. Probabilistic diagnosis using a reformulation of the INTERNIST–1/QMR knowledge base: II. Evaluation of diagnostic performance. *Methods of Information in Medicine*, 30(4):256–267, 1991.

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

[9] Agnieszka Oniśko and Marek J. Druzdzel. Effect of imprecision in probabilities on Bayesian network models: An empirical study. In *Working notes of the European Conference on Artificial Intelligence in Medicine (AIME-03): Qualitative and Model-based Reasoning in Biomedicine*, Protaras, Cyprus, October 18–22 2003.

[10] Agnieszka Oniśko, Marek J. Druzdzel, and Hanna Wasyluk. Extension of the Hepar II model to

multiple-disorder diagnosis. In S.T. Wierzchoń M. Kłopotek, M. Michalewicz, editor, *Intelligent Information Systems,* Advances in Soft Computing *Series*, pages 303–313, Heidelberg, 2000. Physica-Verlag.

[11] Agnieszka Oniśko, Marek J. Druzdzel, and Hanna Wasyluk. Learning Bayesian network parameters from small data sets: Application of Noisy-OR gates. *International Journal of Approximate Reasoning*, 27(2):165–182, 2001.

[12] Agnieszka Oniśko, Marek J. Druzdzel, and Hanna Wasyluk. An experimental comparison of methods for handling incomplete data in learning parameters of Bayesian networks. In S.T. Wierzchoń M. Kłopotek, M. Michalewicz, editor, *Intelligent Information Systems,* Advances in Soft Computing *Series*, Heidelberg, 2002. Physica-Verlag (A Springer-Verlag Company). 351–360.

[13] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1988.

[14] Malcolm Pradhan, Max Henrion, Gregory Provan, Brendan del Favero, and Kurt Huang. The sensitivity of belief networks to imprecise probabilities: An experimental investigation. *Artificial Intelligence*, 85(1–2):363–397, August 1996.

[15] Frank Rijmen. Bayesian networks with a logistic regression model for the conditional probabilities. *International Journal of Approximate Reasoning*, in press.

[16] M.A. Shwe, B. Middleton, D.E. Heckerman, M. Henrion, E.J. Horvitz, H.P. Lehmann, and G.F. Cooper. Probabilistic diagnosis using a reformulation of the INTERNIST–1/QMR knowledge base: I. The probabilistic model and inference algorithms. *Methods of Information in Medicine*, 30(4):241–255, 1991.

[17] Linda C. van der Gaag and Silja Renooij. Analysing sensitivity data from probabilistic networks. In *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference (UAI-2001)*, pages 530–537, San Francisco, CA, 2001. Morgan Kaufmann Publishers.

Methods for Representing Bias in Bayesian Networks

Eric Carlson, Sean Guarino, Jonathan Pfautz
Charles River Analytics, Inc.
625 Mount Auburn Street
Cambridge, MA 02138

Abstract

Bias is intrinsic to observation and reasoning in both humans and automated systems. Bayesian Belief Networks (BBNs) are well suited for representing these biases and for applying bias models to improve reasoning practices, but there are a number of different ways that bias can be represented and integrated into reasoning processes using BBNs. In this paper, we describe a number of methods to model biases using BBNs and discuss the strengths and weaknesses of each method.

1. INTRODUCTION

Bias is intrinsic to observation and reasoning. Though the concept carries connotations of human judgment, bias also applies to automated systems, introduced by the limitations of their capabilities. Reasoning about information that includes bias (i.e., processed information, whether from human of machine) requires reasoning about the information, itself, and about the biases that influenced it. Humans do this naturally. In rich human-to-human interactions, each person derives an understanding of the biases involved from shared context and estimates of the other's attitudes and beliefs. In other circumstances, such as shallow person-to-person interactions (e.g., reading a restaurant review from an unknown person) or interactions involving automated processes (e.g., getting directions from a GPS; incorporating human reports into an automated decision aide; integrating contributions from multiple sensor systems in a data fusion system), biases and their influences need to be made explicit. As Hastie & Dawes (2001) argue, incorporating an explicit model of biases and their influences into reasoning processes can lead to more robust and accurate reasoning in both humans and automated systems.

Bayesian Belief Networks (BBNs) are well suited for modeling biases in automated processing systems and decision aides. Many factors contribute to bias, interacting in a complex manner with each other and with the overall bias. BBNs represent the type of probabilistic influences and causal relationships required to capture this interaction (Pearl & Russell, 2000; Pearl, 2001). Furthermore, the graphical nature of BBNs further supports the expression of these relationships by providing an intuitive method to capture contributing factors and influences. In addition to providing an applicable modeling approach for capturing biases, BBNs are already applied in many fields where consideration of biases has the potential to make significant contributions to performance and realism, such as military intelligence (Koelle et al., 2006; Pfautz et al., 2005a; Pfautz et al., 2005b), medical diagnostics (Kononenko, 1993; Parmigiani, 2002; Nikovski, 2000), and human behavior modeling (Guarino et al., 2006; Hudlicka & Pfautz, 2002; Neal Reilly et al., 2007; Pfautz & Lovell, 2008).

To advance the incorporation of bias models in these fields and others, in this paper we discuss the role of biases in the decision making process (which includes, for our purposes here, observation, reasoning, and decision selection), several ways bias can be modeled using BBNs, and the benefits and drawbacks of each of these methods.

2. BACKGROUND

The study of biases to date largely focuses on cognitive biases. Several attempts have been made to categorize different types of bias and to identify how they affect the decision-making process. One method for classification is to look at the source of the bias, for instance, dividing uncertainty into forms that come from computational models as opposed to human interpretation (Schunn, Kirschenbaum, & Trafton, 2003). Another method is to examine the use of bias and uncertainty in the decision-making process, resulting in categories, which has resulted in categories such as *executional uncertainty*, *goal uncertainty*, and *environmental uncertainty* (Yovits & Abilock, 1974). Another set of classifications developed by Lipshitz and Strauss (1996) divides forms of uncertainty into *inadequate understanding*, *lack of information*, and *conflicted alternatives.* Similar taxonomies were developed by Schunn et al. (2003) and Klein (1998). These taxonomies can prove to be useful in attempts to develop descriptive models of human reasoning. For example, Lipshitz & Straus (1996) discuss

five strategies for reasoning under uncertainty: 1) reduce uncertainty by collecting more information; 2) use assumptions to fill in gaps of knowledge; 3) weigh pros and cons; 4) forestall; and 5) suppress uncertain information. While these classifications of uncertainty and an understanding of biases they introduce to decision-making have been useful in the development of models of human behavior, they may not generalize to other types of biases.

3. ROLE OF BIAS

For the purpose of incorporating consideration of bias into reasoning process, we are concerned with bias in two separate roles. First, because bias impacts the creation of the products of observations and reasoning processes, it must be accounted for in the *interpretation* of those products. Limitations, methods, and, in the case of humans, preferences and cognitive biases introduce a systematic modification into an observed product. This modification must be identified and defined to properly reason based on these products. Elaborating on the earlier example, consider a negative review of a French restaurant written by someone who dislikes French food. Whether he is cognizant of this influence or not, the product of his observation—the review—incorporates his pre-existing preference. To reason based on this review, anyone reading it needs to recognize and correct for the preferences of the reviewer. Automated systems may not have personal preferences, but their technical limitations can introduce similar biases. Consider a sensor that detects the presence of humans based on heat signatures. Because readings are based on the contrast between the person and the ambient temperature, this sensor has a higher occurrence of false negatives when the temperature is above body temperature. So, a reading showing no people present on a 100°F day may be disingenuous because it is the product of both the reading and the hidden bias introduced by its technical limitations. As with the previous example of human bias, the consumer of this automated report—human or automated system—must reason about both the contributing bias and the information, itself, to accurately use the product.

Second, bias impacts the *reasoning* process applied to make decisions based on information products. The consumer introduces its own systematic modification of the information based both on its own biases and on the perceived biases incorporated in the product. For example, the analysis system using reports from the heat sensor may incorporate the fact that it does not function if the temperature is over 100°F , and disregard the sensor's information products on a particularly hot days. Similarly, the analysis may favor one sensor type over another for gathering specific information, regardless of specific conditions (e.g., an analysis system may trust a radar over an eye witness due to a bias against non-technical sources). In this role, bias is not considered solely in the context of information production (though this may be considered); these biases consider how the information is being used and the reasoning processes involved.

These two roles are cyclic, as the results of a reasoning process can be viewed as its own information product. If there are known biases in that product, an estimate of those biases may become an element in a new consumer's reasoning processes, alongside other reasoning biases of the consumer. When the information product being interpreted pertains to an observable truth (e.g., a sensor detecting some object), understanding the influence of bias allows the consumer to determine the accuracy of the product and to integrate that accuracy information into its own reasoning processes. When the product pertains to a subjective belief or assessment (e.g., an opinion about a restaurant), understanding the contributing biases allows the consumer to determine how to integrate those biases with its own biases.

These two roles comprise use cases for bias models, each with their own concerns motivating different design decisions. In the *interpretation* role, a model of bias can serve as a mechanism to correct for biases. Here, the details of the sources of those biases may not necessarily be important. Rather, it is important to correct for errors caused by biases. In the *reasoning* role, a model can be used to self-regulate against the introduction of additional biases, as well as to increase the accuracy of the consumer's estimation of biases contributing to a product, which allows information to be incorporated into the consumers own reasoning at the highest fidelity possible. Here, the details of the sources of those biases may be extremely important, as different meta-information and information may have a direct influence in the reasoning process.

4. THE STRUCTURE OF BIAS

As a concept, bias is closely related to meta-information. Meta-information is information about information. That is, information that serves to qualify and give context to other information. For example, if a sensor reading is information, the fact "the reading is two weeks old" is meta-information—information about the report. For a more extensive discussion of meta-information, see (Guarino et al., 2006). Whereas meta-information is a statement of fact ("the report is old"), bias is the effect meta-information has on observations and reasoning processes ("because the report is old, its contents are probably inaccurate"). Thus, information types can be divided into three levels:

1) the information, itself (e.g., the contents of the report)
2) meta-information (e.g., information about the report)
3) biases (e.g., the impact information *about* the report—the meta-information—has on the information *in* the report)

Biases are derived from meta-information by combining that meta-information with elements of the information.

For example, a two week old sensor reading showing the location of people in an open setting would not convey their current location with high confidence, while a two week old sensor reading showing the location of buildings would represent their current position with a high degree of certainty. So, in this example, the bias ("the information in the reading is wildly inaccurate") is derived from a factor of the information ("people move frequently") combined with meta-information ("the report is ten days old"). This same logic holds for subjective assessments. In the restaurant review example,

- Meta-information: The reviewer hates French food
- Information: The restaurant is French
- Bias: The reviewer was predisposed to hate the restaurant, regardless of its quality

These definitions of information types and the derivation of bias are the basis for the structure of our bias models.

5. BIAS MODELS

In this section, we present a number of ways to model bias, and we discuss the advantages and disadvantages of each model in light of the roles of bias (see section 3) and additional concerns about model use and creation. Bias models vary along two dimensions: the level of detail expressed about the bias and the level of integration with the reasoning model to which it is meant to contribute.

5.1 IMPLICIT BIAS MODEL

The implicit bias model does not contain a representation of the bias in its structure. Instead bias is expressed in the relationship between nodes of the existing elements of the model. Insomuch as it exists anywhere, the bias exists in each node's Conditional Probability Tables (CPTs). The effect this bias exerts on the product of the model—the observation, decision, behavior, etc.—is a change in the beliefs of the nodes. The bias, itself, is not explicitly represented separate from the state information of the model. For example, see Figure 5-1, an implicit bias model of our previous heat sensor example.

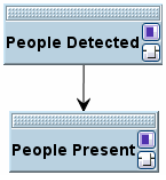


Figure 5-1: Implicit bias model structure of the heat sensor example. Bias is represented only in the CPTs.

The sole factor represented as contributing to whether people are present is the number of people detected by the sensor. The bias in this model is expressed as uncertainty in the outcome. For positive readings, the likelihood of people being present is high. Because there are conditions that can increase the likelihood of false negatives, though, a negative reading leads to a lower certainty of people not being present (see Figure 5-2).

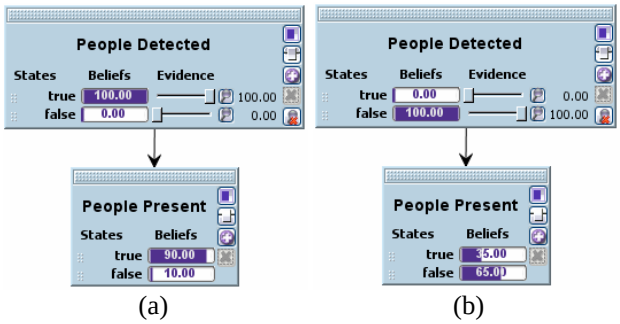


Figure 5-2: (a) left, shows the high belief that people are present based on a positive reading of the heat-based sensor; (b), right, shows a less certain belief that people are not present based on a negative reading of the same sensor. The bias is reflected in the increased uncertainty due to the possibilities of false negatives.

The implicit bias model reflects the simplest case. Though it does reflect the reality of the situation, this model is insufficient in most other ways. Because elements that contribute to the bias (i.e., meta-information) are not explicitly represented, the bias is reflected in a permanent change in confidence rather than reflecting specific conditions (e.g., because the ambient temperature is not explicitly represented, the confidence cannot change based on the specific value of that variable). Instead, this model merely represents that bias is possible in the reasoning process. This model may be sufficient for representing bias while interpreting data because the value of the relevant meta-information may not be available to the consumer. However, because it does not explicitly describe the contributing factors and applies the bias as a consistent change in certainty rather than on a case-by-case basis, it is ineffective at providing a nuanced bias model for reasoning.

5.2 INTEGRATED BIAS MODEL

In an integrated model, the factors that contribute to bias (i.e., meta-information) are explicitly represented as nodes in the network and are fully integrated into the model of the observation, reasoning process, behavior, etc. The bias—the effect of this meta-information—is still contained in the CPTs. Like the implicit model, there is a bias in the computational process, but that bias is not explicitly represented as a node in the BBN. Figure 5-3 expands Figure 5-1 into an integrated model by adding Ambient Temperature as an input node.

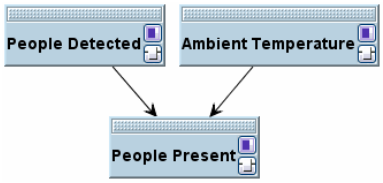
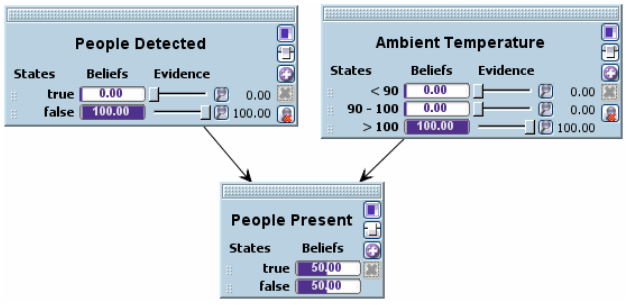
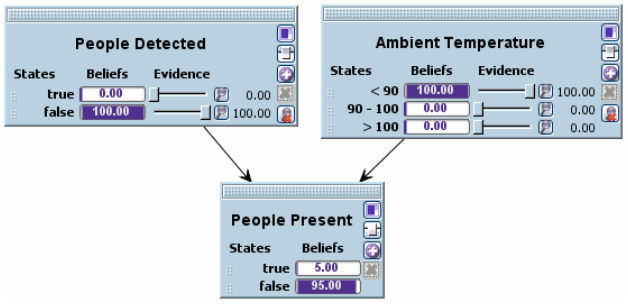


Figure 5-3: An integrated bias model of the heat sensor example. Meta-informational factors are represented. Bias is represented in the CPTs.

This inclusion of factors that moderate biases allows the bias model to account for the exact value of relevant meta-information, allowing the bias to change dynamically (see Figure 5-4). Furthermore, because each factor is expressed independently, their combined effect on the reasoning process can be nuanced.



(a)



(b)

Figure 5-4: Integrated bias model of the heat sensor example. In (a), a high ambient temperature increases uncertainty. In (b), a low ambient temperature decreases uncertainty. The bias reacts in real-time to conditions, increasing accuracy of the model.

In an integrated bias model, factors contributing to bias are explicitly expressed, so these models are more accurate, and, therefore, better than implicit models in an interpretation role. However, as in the implicit model, the effect of these factors is still captured fully in the CPTs. For this reason, expansion of the model is difficult, as additions could require significant modifications to those CPTs. Therefore, in a reasoning role it is difficult to adapt parts of an integrated bias model for reuse in a larger reasoning model.

5.3 CONSOLIDATED UNKNOWN BIAS MODEL

In a consolidated unknown bias model, bias is expressed as a single node in the network, with connections to each of the nodes in the network. This single node is a "black box" meant to represent the amount of bias in the model with no concern for the cause of the bias (note: this node could be a placeholder for bias calculated using the standalone bias model discussed in sections 5.5 and 5.6). For an example of a consolidated unknown bias model, see Figure 5-5.

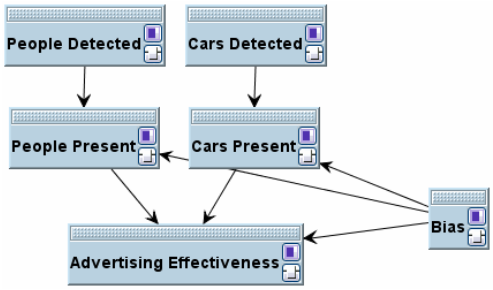


Figure 5-5: A consolidated unknown bias model, where the strength of the bias present is represented by a single node, which connects to all elements of the reasoning model.

This model does contain a mechanism to express bias in every part of the model, but it makes a large assumption about the distribution of that bias. The effect bias has on each element is expressed in the CPTs, which means that the specific effects of the bias strength is individual to each node, but the strength is shared. This model does represent the effect of bias on a gross level, so it can be used somewhat in an interpretation role, albeit with lower fidelity since all biases are expressed in a single dimension. The effect of the bias is hidden in the CPTs, and the factors that contribute to the bias are completely unstated, so in a reasoning role biases cannot be utilized by addition elements of a reasoning model.

5.4 DISTRIBUTED UNKNOWN BIAS MODEL

The distributed unknown bias model represents bias as a number of "black boxes", each having an effect on one or more elements of the reasoning model. Again, as black boxes, the factors contributing to each bias are not explicit. Bias nodes provide an overall representation of the biases in the reasoning components to which they are attached. For an example, see Figure 5-6.

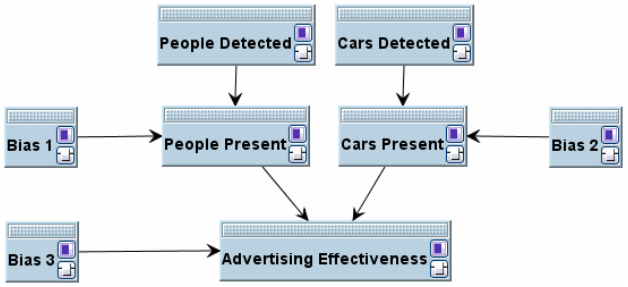


Figure 5-6: A distributed unknown bias model, where bias is represented as a number of unknowns, each connected to elements of the reasoning model.

Distributed unknown bias models are superior to consolidated unknown bias models because they express a more nuanced situation reflecting the susceptibility of various elements of the model to different biases. The bias nodes play a similar role to meta-information nodes in an integrated bias model, but, as black-box bias modules, they consolidate all factors contributing to a particular bias into a single node. In an interpretation role, these models are more useful than implicit bias models because at least some gauge of the strength of bias active in each element is present. However, unlike the integrated bias model, the meta-information factors that affect their strength are unknown. This reduces the already limited ability of bias factors in distributed unknown bias models to be integrated into an external reasoning model. Unlike the models representing meta-information factors explicitly, the ability to add factors is not a concern because they are aggregated together in a single node, so no CPTs need to be changed. However, without expressing the composition of the bias, the bias strengths and relationships are highly subjective.

5.5 STANDALONE BIAS MODEL

A standalone bias model expresses bias in an independent model separate from the reasoning model. This is distinct from the integrated model where factors are represented but are integrated with the reasoning model itself. The measure of bias resulting from this model can then be applied to the reasoning model, filling the black-box need of the consolidated or distributed bias model, or used alone. Bias is expressed explicitly as a single node. Each element in a reasoning model where bias is a factor would require an independent bias model. The mechanism by which each factor contributes to bias is hidden in the CPTs. For an example of a standalone bias model, in the heat-based human detector the meta-information factor “Ambient Temperature” could be expressed (alongside any other relevant factors) as explicit nodes. The effect that each factor has (i.e., that high temperature increases the uncertainty of negative readings) is still expressed only in the CPTs. This example is depicted in Figure 5-7.

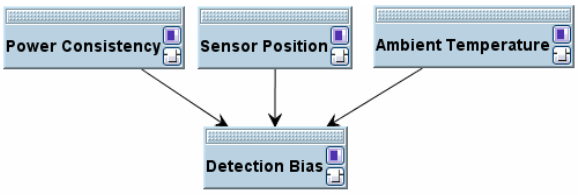


Figure 5-7: A standalone bias model of detection bias for a heat-based person detector. The product of this standalone model could then be applied in a reasoning model.

Like the integrated model, because standalone bias models represent the contribution of each of a set of factors to a bias explicitly, these models can dynamically capture bias, providing greater accuracy. Expressing factors in a separate model allows them to easily be applied as a factor in a large or frequently changing model. For this reason, standalone bias models excel in circumstances where a bias model might be applied independently at multiple points in a reasoning process.

For example, consider a data fusion application that receives sporadic inputs from a host of sensors. Rather than use a single monolithic model that integrates information from all sensors, standalone bias models could be used to dynamically assemble a model that represents only those sensors that are currently active. Because the majority of the sensors are silent at any given time, this improves the efficiency of bias application in such conditions. However, this autonomy has a tradeoff in that bias is consolidated into a single metric resulting in the influence of specific pieces of meta-information having limited nuance in their effect on the reasoning process. Furthermore, an element or even a network fragment might be repeated in multiple standalone models leading to wasteful repetitive computation. Nevertheless, due to the explicit representation of meta-informational factors and simple portability, this type of model applies well in both interpretation and reasoning roles.

5.6 FULLY ENUMERATED STANDALONE BIAS MODEL

Fully enumerated standalone bias models explicitly represent both the meta-information that causes the bias and the element that defines how that meta-information contributes to bias (as discussed in Section 4). Rather than a single model for each bias type as with the standalone bias model, fully enumerated standalone bias model have a single model for each element of the information that, when paired with meta-information, could introduce a bias. These models express all factors contributing to bias and the bias itself as elements in the network, rather than being contained in the CPTs. For an example of fully enumerated standalone bias models, see Figure 5-8.

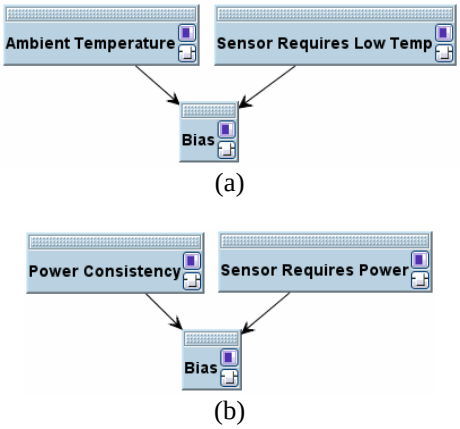


Figure 5-8: Fully enumerated standalone bias models for (a) bias related sensors whose performance is affected by temperature, and (b) bias related to sensors whose performance is affected by power supply.

Similar to the way standalone bias model can be applied dynamically based on the biases present, fully enumerated standalone bias models can be applied based on the definition of the system creating the product. So, a system using these needs a model for each possible property of the data sources. It can then apply them based on the definition of each source. For example, in a fusion system designed to dynamically calculate bias for any configuration of sensors, a bias model could be automatically assembled for each sensor based on the operating characteristics of that sensor. The heat sensor, defined as requiring low temperature, would incorporate biases related to that requirement. Because these networks determine the bias introduced by each factor separately, their integration into a reasoning process can be more nuanced than representations that consolidate bias into a single measure. This, along with the transparency of contributing factors, makes them ideal in a reasoning role.

6. CONCLUSIONS

There are numerous ways to represent bias as a BBN, each of which has its own strengths and weaknesses. Models of bias provide a mechanism to correct for bias to increase accuracy and to integrate biased information into human and automated reasoning processes. The most advantageous form of model for a particular situation depends on its intended use.

By systematically examining the composition of bias, we have identified factors in its composition. The various model types we discussed make use of this definition by incorporating various factors at a range of fidelities, making specific elements more or less accessible. Additionally, we have defined two separate roles bias can play in reasoning processes. These roles form the basis for use cases, which we have used to evaluate each of the types of models. Of the models discussed, the more nuanced the application of bias to elements that contributed to the production of information, the greater the benefit in accurately interpreting the product of reasoning processes without introducing additional biases. To reason based on those products, those models that include the greatest level of detail and autonomy for factors that contribute to bias can be more easily and accurately integrated into reasoning processes.

7. DISCUSSION

This set of bias representations encapsulates a significant range of capabilities and tradeoffs. Among the most prominent difference between these representations is the degree of specificity about the sources of bias. In certain applications, like accounting for bias from a technical sensor, these bias factors can be easily identified and described. In others, like accounting for bias in human reasoning, these sources are obscured and can only be hypothesized through intense effort, and are largely unverifiable. In light of these impediments, going forward we need to determine what guidelines could be established to govern the applicability of different styles. How can uncertainty about the causes of bias be mitigated? Is there a way to create representations that don't incorporate unspecified sources of bias, but that are applicable in situations where those sources are vaguely defined? Or, are there ways to use black box bias measures without fully sacrificing the attribution that identifying specific sources provides? Is this attribution of bias to particular sources necessarily important (e.g., for accountability, trust)? What conditions of use make attribution important (e.g., frequent updates, logic exposed to the user)? The complexity of specificity results, too, in a gain in precision in the end bias measure. Can factors contributing to bias be calculated precisely enough to warrant this precision in the end product?

Acknowledgements

The authors would like to express their deepest gratitude to the subject matter experts who contributed to our understanding of biases. Additionally, we would like to thank Dr. Greg Zacharias for funding our work on Bayesian Belief Networks.

References

Guarino, S., Pfautz, J., Cox, Z., & Roth, E. (2006). Modeling Human Reasoning About Meta-Information. In *Proceedings of 4th Bayesian Modeling Applications Workshop at the 22nd Annual Conference on Uncertainty in AI: UAI '06*. Cambridge, Massachusetts.

Hastie, R. & Dawes, R. M. (2001). *Rational Choice in an Uncertain World: The Psychology of Judgment and Decision-Making*. London, UK: Sage Publications.

Hudlicka, E. & Pfautz, J. (2002). Architecture and Representation Requirements for Modeling Effects of Behavior Moderators. In *Proceedings of Proceedings of 11th Conference on Computer-Generated Forces - Behavior Representation*. Orlando, FL.

Klein, G. A. (1998). *Sources of Power: How People Make Decisions*. Cambridge, MA: MIT Press.

Koelle, D., Pfautz, J., Farry, M., Cox, Z., Catto, G., & Campolongo, J. (2006). Applications of Bayesian Belief Networks in Social Network Analysis. In *Proceedings of 4th Bayesian Modeling Applications Workshop at the 22nd Annual Conference on Uncertainty in AI: UAI '06*. Cambridge, Massachusetts.

Kononenko, I. (1993). Inductive And Bayesian Learning in Medical Diagnosis. *Applied Artificial Intelligence*, 7(4), 317-337.

Lipshitz, R. & Strauss, O. (1996). How Decision-Makers Cope With Uncertainty. In *Proceedings of Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting-1996*, (pp. 189-193). Philadelphia: Human Factors Society.

Neal Reilly, W. S., Bayley, C., Koelle, D., Marotta, S., Pfautz, J., & Keeney, M. (2007). *Culturally Aware Agents for Training Environments (CAATE): Final Report.* (Rep. No. R070101). Cambridge, MA: Charles River Analytics, Inc.

Nikovski, D. (2000). Constructing Bayesian Networks for Medical Diagnosis From Incomplete and Partially Correct Statistics. *IEEE Transactions on Knowledge and Data Engineering*, 12(4), 509-516.

Parmigiani, G. (2002). *Modeling in Medical Decision Making: A Bayesian Approach*. John Wiley and Sons.

Pearl, J. (2001). *Causality: Models, Reasoning, and Inference*. Cambridge Univ Press.

Pearl, J. & Russell, S. (2000). *Bayesian Networks.*

Pfautz, J., Fouse, A., Roth, E., & Karabaich, B. (2005a). Supporting Reasoning About Cultural and Organizational Influences in an Intelligence Analysis Decision Aid. In *Proceedings of International Conference on Intelligence Analysis*. McLean, VA.

Pfautz, J. & Lovell, S. (2008). Methods for the Analysis of Social and Organizational Aspects of the Work Domain. In A. Bisantz & C. Burns (Eds.), *Applications of Cognitive Work Analysis*. Lawrence Erlbaum Associates.

Pfautz, J., Roth, E., Bisantz, A., Fouse, A., Madden, S., & Fichtl, T. (2005b). The Impact of Meta-Information on Decision-Making in Intelligence Operations. In *Proceedings of Human Factors and Ergonomics Society Annual Meeting*. Orlando, FL.

Schunn, C. D., Kirschenbaum, S. S., & Trafton, J. G. (2003). The Ecology of Uncertainty: Sources, Indicators, and Strategies for Information Uncertainty. http://www.au.af.mil/au/awc/awcgate/navy/nrl_uncertainty_taxonomy.pdf [On-line].

Yovits, M. C. & Abilock, J. (1974). A Semiotic Framework for Information Science Leading to the Development of a Quantitative Measure of Information. In *Proceedings of 37th American Society for Information Sciences Meeting*, (pp. 163-168).