Risk Analysis, Vol. 0, No. 0, 2021 DOI: 10.1111/risa.13759

BARD: A Structured Technique for Group Elicitation of Bayesian Networks to Support Analytic Reasoning

Erik P. Nyberg,[1] Ann E. Nicholson,[1,*] Kevin B. Korb,[1] Michael Wybrow,[1] Ingrid Zukerman,[1] Steven Mascaro,[2] Shreshth Thakur,[1] Abraham Oshni Alvandi,[1] Jeff Riley,[1] Ross Pearson,[1] Shane Morris,[3] Matthieu Herrmann,[1] A.K.M. Azad,[1] Fergus Bolger,[4] Ulrike Hahn,[5] and David Lagnado[6]

In many complex, real-world situations, problem solving and decision making require effective reasoning about causation and uncertainty. However, human reasoning in these cases is prone to confusion and error. Bayesian networks (BNs) are an artificial intelligence technology that models uncertain situations, supporting better probabilistic and causal reasoning and decision making. However, to date, BN methodologies and software require (but do not include) substantial upfront training, do not provide much guidance on either the model building process or on using the model for reasoning and reporting, and provide no support for building BNs collaboratively. Here, we contribute a detailed description and motivation for our new methodology and application, Bayesian ARgumentation via Delphi (BARD). BARD utilizes BNs and addresses these shortcomings by integrating (1) short, high-quality e-courses, tips, and help on demand; (2) a stepwise, iterative, and incremental BN construction process; (3) report templates and an automated explanation tool; and (4) a multiuser web-based software platform and Delphi-style social processes. The result is an end-to-end online platform, with associated online training, for groups without prior BN expertise to understand and analyze a problem, build a model of its underlying probabilistic causal structure, validate and reason with the causal model, and (optionally) use it to produce a written analytic report. Initial experiments demonstrate that, for suitable problems, BARD aids in reasoning and reporting. Comparing their effect sizes also suggests BARD's BN-building and collaboration combine beneficially and cumulatively.

KEY WORDS: Delphi process; probabilistic graphical models; probabilistic reasoning

1. INTRODUCTION

In many complex real-world situations, problem solving and decision making require effective reasoning about causation and uncertainty. The effectiveness of human reasoning in these cases is limited: It may handle simple, quasi-linear cases well, but in complex or nonlinear cases, it is notoriously prone to confusion and error (Hahn & Harris, 2014; Kahneman, Slovic, & Tversky, 1982; Newell, Lagnado, & Shanks, 2015). One way to handle such

[1]Department of Data Science & AI, Monash University, Melbourne, Australia.
[2]Bayesian Intelligence Pty Ltd, Melbourne, Australia.
[3]Automatic Studio Pty Ltd, Melbourne, Australia.
[4]Strathclyde Business School, University of Strathclyde, Glasgow, UK.
[5]Department of Experimental Psychology, University College London, London, UK.
[6]Department of Psychological Sciences, Birkbeck, University of London, London, UK.
[*]Address correspondence to Ann E. Nicholson, Department of Data Science & AI, Monash University, Clayton VIC 3800, Australia; tel: +61 448 019 439; ann.nicholson@monash.edu.

reasoning more effectively is to employ Bayesian networks (BNs) (Korb & Nicholson, 2011; Pearl, 1988), which are an artificial intelligence (AI) technology that models uncertain situations, representing them clearly for the user and making complex calculations quickly and accurately on demand, thus supporting better probabilistic and causal reasoning and decision making.

BNs have been deployed for this purpose in diverse domains such as medicine (Flores, Nicholson, Brunskill, Korb, & Mascaro, 2011; Sesen, Nicholson, Banares-Alcantara, Kadir, & Brady, 2013), education (Nicholson et al., 2001), engineering (Bayraktar & Hastak, 2009; Choi, Joo, Cho, & Park, 2007; Misirli & Bener, 2014), reliability assessment (Langseth & Portinale, 2007; Sigurdsson, Walls, & Quigley, 2001), surveillance (Mascaro, Nicholson, & Korb, 2014), the law (Fenton, Neil, & Lagnado, 2013; Lagnado & Gerstenberg, 2017), weather forecasting (Boneh et al., 2015), and the environment (Chee et al., 2016). Furthermore, BNs have been used to analyze common fallacies in informal logic (Korb, 2004), analyze and assess a variety of arguments in criminal law, thus exposing some common errors in evidential reasoning (Fenton et al., 2013; Lagnado, Fenton, & Neil, 2013), analyze human difficulties with reasoning under uncertainty (Hahn, 2014; Hahn & Oaksford, 2006), and proposed as a general structured method for argument analysis (Korb & Nyberg, 2016).

In this article, we contribute a detailed description and motivation for our new methodology and application, Bayesian ARgumentation via Delphi (BARD), which combines BNs with a Delphi social process: a systematic method for combining multiple perspectives in a democratic, reasoned, iterative manner (Linstone & Turoff, 1975). The initial motivation for BARD was to extend the use of BNs to a new domain: intelligence analysis. Development began as part of the Crowdsourcing Evidence, Argumentation, Thinking and Evaluation (CREATE) program funded by the Intelligence Advanced Research Projects Activity (IARPA)[1]. The CREATE program sought to develop, and experimentally test, systems that use crowdsourcing and structured analytic techniques to improve analytic reasoning, including to help users better understand the evidence and assumptions that support or conflict with conclusions. CREATE's secondary aim was to help users to better communicate their reasoning and conclusions. This included meeting the standards for high-quality analytic reports outlined in the Intelligence Community Directive 203 (ICD-203) (Clapper, 2015). Furthermore, as CREATE demanded, BARD aims to improve the quality of analysis and communication using BNs without requiring users to have prior BN expertise or the assistance of a BN expert. This makes BNs accessible to the uninitiated, so those outside the BN community can benefit from them.

BNs take advantage of the natural ability of humans to reason and build causal models about the world (Bramley, Dayan, Griffiths, & Lagnado, 2017; Lagnado & Sloman, 2004, 2006; Sloman & Lagnado, 2015). However, for domain experts to construct their own BNs, current software has major deficiencies: (1) It usually requires substantial upfront training, not included in the software, (2) it does not provide much guidance on the model building process, or (3) on using the resulting model for reasoning and reporting, and (4) it does not provide any support for collaboratively building BNs. BARD addresses each of these deficiencies by integrating the following key novel features:

(1) *Short, High-Quality E-courses, Tips, and Help on Demand:* BARD provides compressed, high-quality, training to allow novices to start using the system as soon as possible, and then receive further help as needed. All key elements of the BARD approach are condensed into four hours of short, interactive, modular e-courses. These are augmented by embedded help components that include training problems with ideal solutions, optional product tours, context-specific tips, and guidance on what to do next.
(2) *A Stepwise, Iterative, and Incremental BN Construction Process:* BARD breaks down a given task into six steps that are performed by the analysts: (1) premodeling exploration of the problem to be solved, (2–4) building the components of the BN, (5) exploring the BN's reasoning on specific scenarios, and (6) report writing with BARD's support. However, progress need not be linear: BARD encourages analysts to incrementally and iteratively build their individual BNs, and seek regular feedback through communication with other group members and the facilitator.
(3) *Analytical Report Templates and an Automated Explanation Tool (AET):* BARD guides verbal reporting with an analytical template, designed to elicit relevant points in a logical

[1] https://www.iarpa.gov/

and thorough way that is consistent with general good reasoning guidelines (e.g., Clapper, 2015). BARD also autogenerates from the BN model many key points, in English, organized according to the same template—such as the diagnosticity of evidence and critical uncertainties—which analysts or the facilitator can easily incorporate into their reports.

(4) *A Multiuser Web-Based Software Platform and Delphi-Style Social Processes:* Analysts in small groups, optionally assisted by a facilitator, are guided through a structured Delphi-style elicitation protocol to consider and represent their problem-relevant knowledge in a causal BN augmented by descriptive annotations. BARD provides tools (described in Section 3) to assist with the elicitation of BN structure and parameters, with review and consensus building in the group, and with evaluation of the results. Delphi protocols are designed to avoid common pitfalls of group deliberation, e.g., overweighting the first opinion expressed. Analysts are required to first develop an answer on their own, in a private phase, then BARD shows other group members' contributions after the analyst has published his or her initial attempt. This approach maximizes the diversity of answers from which the group starts its work, then encourages analysts to improve their own answers, often by incorporating good features proposed by others and converging on a consensus. Other major features of Delphi utilized by BARD are anonymity and moderated discussion, both of which should help groups avoid being unduly influenced by high status or opinionated individuals rather than the knowledgeable.

In Section 2, we provide background information on the two existing techniques that BARD combines, BNs and Delphi, and their previous applications to improve reasoning. In Section 3, we present our detailed description and motivation for BARD's many features. BARD offers unique benefits for the group elicitation of influence diagrams (i.e., causal graphs) or full BNs (i.e., with probabilities), and (where applicable) improved reasoning and reporting about the situations modeled.

Its efficacy is supported by experimental results summarized in Section 4 and reported in detail elsewhere. In Section 5, we outline directions for future development.

2. BACKGROUND

In this section, we summarize the relevant background information about BNs, their previous applications to overcoming reasoning errors, techniques for eliciting them from experts, and algorithms for explaining them. We also summarize the characteristic features of Delphi elicitation protocols for group decision making, and the errors they help to overcome.

2.1. BNs

BNs are directed, acyclic graphs whose nodes represent the random variables of a problem, and whose directed links (arrows) represent direct probabilistic dependencies between the nodes they connect (Pearl, 1988; Korb & Nicholson, 2011). In causal BNs, these arrows also represent direct causal influence. Each node at the tail of an arrow is called a *parent* of the *child* node at the head of the arrow. The relationship between each child and its parents is quantified for discrete variables (i.e., those with a finite number of possible states) by a *conditional probability table* (*CPT*) associated with the child node, which specifies the probability of each child state given each combination of parent states. BNs thus provide a compact representation of the full joint probability distribution of their variables. Users can specify the values of any combination of variables in a BN, usually on the basis of observed evidence e. There are several efficient algorithms for propagating this evidence through the network, quickly producing a posterior probability distribution $P(X|e)$ for each of the other variables X, and thus supporting predictive, diagnostic, and explanatory reasoning.

BNs are the culmination of a century of research on models formally representing causal relations, beginning with the work of Sewall Wright in the 1920s and 30s on path models (Wright, 1934). This tradition has given rise to structural equation models, which underwrite much of the formal work in economics, psychology, and the social and biological sciences. It also led to work on discovering causal relations from data, including work by Herbert Simon and Hubert Blalock Jr. on "non-experimental" causal inference (e.g., Blalock Jr, 1964; Simon, 1954). In the 1980s, statisticians and AI researchers developed new techniques for modeling probability distributions, which were codified in Judea Pearl's text *Probabilistic Reasoning in Intelligent Systems* (Pearl, 1988). This work launched BNs as a modeling tool for

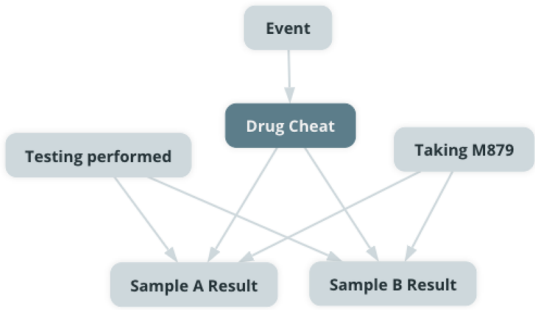
BARD Training Problem: The Drug Cheat

After competing, a proportion of competitors at the Olympics are randomly chosen for testing for the presence of performance enhancing drugs. Here, we'll consider only competitors from three sports: athletic runners, swimmers, and weightlifters. Drug tests conducted in the past indicated that 4% of weightlifters take performance enhancing drugs, while runners are half as likely as weightlifters to take such drugs, and swimmers are half as likely as runners to do so. The testing error rates are 2% false positive and 5% false negative. When an athlete is chosen for drug testing, two samples are taken: the A and the B sample, with the B sample only analyzed if the A sample comes back positive. The probability threshold for being found guilty, which will result in automatic disqualification and a 2 year ban, is 98%.

Consider the scenario of a swimmer, Sam, who is randomly chosen for testing. Sam returns a positive result for the sample A test, and then for the sample B test. Sam claims that the positive test results were not caused by performance enhancing drugs, but by taking a medication, M879, prescribed by her doctor and on the Olympics' approved list. M879 has recently been found to trigger a positive result in the test for performance enhancing drugs. Sam's doctor confirms that Sam did take this medication regularly for a very rare condition. Based on the information and evidence provided, should Sam be found guilty, and hence disqualified and banned for 2 years?

Variable	States
Event	{Weightlifting, Running, Swimming}
Drug Cheat	{True, False}
Testing performed	{True, False}
Sample A Result	{Positive, Negative, No Result}
Sample B Result	{Positive, Negative, No Result}
Taking M879	{Yes, No}

(a) Variables and their states

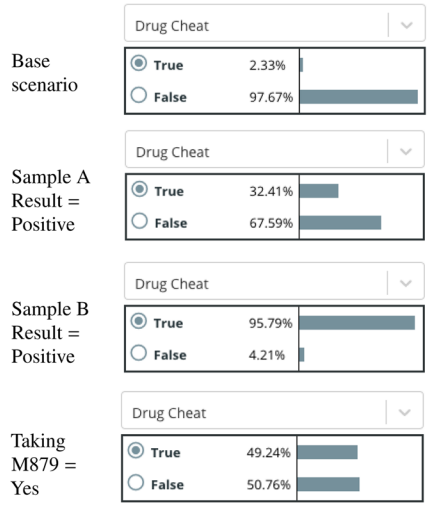


(b) Causal structure

Event	Drug Cheat	
	True	False
Weightlifting	4 %	96 %
Runner	2 %	98 %
Swimmer	1 %	99 %

Drug Cheat	Testing performed	Taking M879	Sample A Result		
			Positive	Negative	No Result
True	True	Yes	98 %	2 %	0 %
True	True	No	95 %	5 %	0 %
True	False	Yes	0 %	0 %	100 %
True	False	No	0 %	0 %	100 %
False	True	Yes	10 %	90 %	0 %
False	True	No	2 %	98 %	0 %
False	False	Yes	0 %	0 %	100 %
False	False	No	0 %	0 %	100 %

(c) CPTs for *Drug Cheat* and *Sample A Result* variables



(d) *Drug Cheat* probability as evidence accumulates

Fig 1. The Drug Cheat Problem, showing the BN variables, its causal structure, two of the CPTs, and the resulting probabilities that Sam the Swimmer is a drug cheat in the base scenario (no evidence) and updated after each additional piece of evidence.

reasoning and decision making under uncertainty, and its theoretical underpinnings as a field of study.

Fig. 1 shows a simple probabilistic reasoning problem that can easily be modeled by a BN, making it easier and quicker to produce precise and correct answers as evidence is updated. This is a training problem within BARD. The *Drug Cheat* variable represents whether an athlete has taken performance enhancing drugs, while *Sample A Result* and *Sample B Result* represent the results of two successive

applications of a test to detect a performance enhancing drug. The variables are all discrete, with their states shown in Fig. 1(a). Fig. 1(b) depicts the BN structure as it appears in the BARD interface; the arrow from the *Event* variable, for example, shows that the probability of *Drug Cheat* = *True* is influenced by the type of sporting event, while the arrows from *Taking M879* indicate that taking this medication affects the probability of a positive test result. The combination of influences on the first test result are quantified in the CPT for *Sample A Result*, while the different drug cheating rates for different sporting events are specified in the CPT for *Drug Cheat*, both shown in Fig. 1(c). The prior probability for a competitor being a drug cheat, *P(Drug Cheat = True)*, is computed to be 2.33%, while the sequence of updated probabilities computed by the BN software for Sam the Swimmer is 32.41% after a positive result for Sample A, jumping to 95.79% after a positive result for Sample B, and finally decreasing to 49.24% in light of new information about taking M879 medication.

The Drug Cheat example illustrates the general calculation advantages of BNs, which help to avoid calculation errors. However, humans are also prone to some more specific reasoning traps, as outlined in Section 2.2, and example problems showing how BNs can help to overcome these can be found in the publications cited there.

2.2. Probabilistic Reasoning Errors

As shown in many studies, human reasoning under uncertainty is fraught with cognitive biases, which result in an incorrect update or utilization of probabilistic information. Some examples are *overconfidence*, i.e., exaggerating the probability of likely events and the improbability of unlikely ones (Lichtenstein, Fischhoff, & Phillips, 1982; Moore & Healy, 2008); *base-rate neglect*, i.e., ignoring objective prior probabilities (Tversky & Kahneman, 1982; Welsh & Navarro, 2012); and *anchoring*, i.e., overreliance on an initial piece of information, the "anchor" (Kahneman et al., 1982).

While many structured representations may assist in the avoidance or mitigation of cognitive biases when analyzing problems, causal BNs are particularly well suited to biases involving probability or causality. By design, BNs require the user to specify explicitly the relevant basic components, in a logically consistent way—thus helping to clarify the user's basic beliefs and identify any missing or inconsistent items. From these beliefs, BNs compute any complex probabilistic consequences quickly and without error or bias. Of course, if some of the basic components elicited are inaccurate, then some of their consequences will be too. The elicitation and validation protocols described in Section 2.4 are designed to promote such accuracy for BNs, and the Delphi protocols described in Section 2.6 are designed to promote accuracy within groups.

The process of modeling reasoning under uncertainty via causal BNs has been shown to help avoid several common human reasoning fallacies, such as base-rate neglect (Korb & Nyberg, 2016); *confusion of the inverse*, i.e., interpreting the likelihood as a posterior (Villejoubert & Mandel, 2002); *the conjunction fallacy*, i.e., assigning a lower probability to a more general outcome than to one of the specific outcomes it includes (Jarvstad & Hahn, 2011); *the jury observation fallacy*, i.e., automatically losing confidence in a "not guilty" verdict when a previous similar conviction by the defendant is revealed (Fenton & Neil, 2000); and *the zero sum fallacy*, i.e., not recognizing when a piece of evidence increases the probability of *both* a hypothesis and its most salient rival (Pilditch, Fenton, & Lagnado, 2019).

In addition, people often make reasoning errors in relation to causality. For example, people often mistake correlation between events for direct causation, even where a hidden common cause is more likely (Gopnik, Sobel, Schulz, & Glymour, 2001; Kushnir, Gopnik, Lucas, & Schulz, 2010; Lagnado & Sloman, 2004; Pearl & Mackenzie, 2018). Causal BNs discourage such mistakes, partly because analysts are required to think about and model direct causal relations explicitly. Two examples of causal reasoning phenomena involving indirect causal connections that are difficult for people to handle, but are correctly captured by causal BNs, are *explaining away*, i.e., when the confirmation of one cause lowers the probability of an alternative cause (Liefgreen, Tešić, & Lagnado, 2018); and *screening off*, i.e., when knowledge of the state of a common cause renders two dependent effects independent of each other (Pearl, 1988).

2.3. BN Tools

Given these benefits, it is not surprising that BNs have been applied to many application areas, as detailed above, in tandem with the development of BN software tools that allow technologists to build, edit, evaluate, and deploy them. Widely used commercial

BN software tools include Hugin,[2] GeNie,[3] Netica,[4] AgenaRisk,[5] and BayesiaLab.[6] In addition, research software and tools include Elvira,[7] R BN libraries,[8] BNT,[9] SamIam,[10] and BayesPy.[11]

2.4. Elicitation of BNs

2.4.1. Data Sets and Experts

Two primary sources of information are commonly employed to learn BN structure and/or parameters: numerical data sets and expert opinions. Given adequate numerical data, machine learning algorithms have been developed to find the sparsest and/or most probable causal BNs that would explain the observed dependencies. Several of these algorithms—e.g., PC (Spirtes, Glymour, & Scheines, 2000), CaMML (O'Donnell, Allison, & Korb, 2006), and the R libraries—are incorporated into standard BN software. Automation helps to overcome the "knowledge-engineering bottleneck" (Korb & Nicholson, 2011) of total reliance on scarce expert resources; however, automated causal structure inferences can be greatly assisted by expert input about some of the likely causal directions (e.g., partially ordering variables in causal tiers).

If necessary, both structure and parameters can be obtained entirely through knowledge elicitation from domain experts. It is common to use multiple opinions rather than relying on only one source, with the hope of obtaining a larger pool of information and more reliability when the majority or average opinion is used—and perhaps benefiting from discussion (see Section 2.6). Since domain experts usually have little BN expertise, they usually need to be assisted by a BN expert, who may also take the role of a moderator who leads discussion. Group elicitation has been particularly common in some domains, such as reliability assessment (Langseth & Portinale, 2007; Sigurdsson et al., 2001) and environmental research (Chee et al., 2016).

[2]https://www.hugin.com/
[3]https://www.bayesfusion.com/
[4]https://www.norsys.com/
[5]https://www.agenarisk.com/
[6]http://www.bayesia.com/
[7]http://leo.ugr.es/elvira/
[8]http://www.bnlearn.com/
[9]https://github.com/bayesnet/bnt/
[10]http://reasoning.cs.ucla.edu/samiam/
[11]https://pypi.org/project/bayespy/

BNs can also be constructed with the help of secondary or intermediate sources. For example, concept mapping is a popular technique for relating concepts in graphs, and such maps can be elicited as a preliminary step in building BNs (Novak, 2010). Similarly, argument diagrams are a well-established technique for representing the logical structure of arguments in trees, which could be used to guide the automated construction of corresponding BNs (Wieten, Bex, Prakken, & Renooij, 2019).

2.4.2. Structure Elicitation

The elicitation of causal structure is a relatively underexplored area. Proposed methodologies for BN elicitation recommend proceeding iteratively and and incrementally (Boneh, 2010; Korb & Nicholson, 2011; Laskey & Mahoney, 1997, 2000). More specifically, Korb and Nicholson (2011, Part III) suggest beginning with a small local structure around a target variable of interest, rather than attempting to exhaustively consider every possible factor relevant to the target. Subsequent iterations can pick up a few additional factors at a time, preferably with some form of validation in each iteration (e.g., feedback from an independent expert). Another recommended strategy is to break down complex models into submodels, and reuse common structures or elements when appropriate, dubbed "idioms" (Fenton & Neil, 2000), "templates" (Laskey & Mahoney, 2000), and "network fragments" (Laskey & Mahoney, 1997).

These incremental approaches adapt similar ideas long used in software engineering, such as "spiral prototyping" or "agile model building" (Boehm, 1988), and reusing common local structures is fundamental to "object-oriented" programming (Cox & Novobilski, 1991). Despite this, none of the commercial BN software packages support the structured elicitation of BNs, or these knowledge engineering principles. They simply assume that users understand BN technology, and know how to translate their knowledge of a causal process or argument into a BN.

To apply group elicitation protocols to BN structure, Serwylo (2015) pioneered using online crowdsourcing and automated aggregation, albeit not in the Delphi style. Nicholson, Mascaro, Thakur, Korb, and Ashman (2016) explored Delphi elicitation and automated amalgamation of structure and parameters, and one recent study has used IDEA, which is a Delphi protocol, to elicit causal structure in conceptual

Table I. Mapping verbal probability descriptors to numerical probability ranges, taken from ICD-203 (Clapper, 2015)

Probability Expressions	
Verbal	Numerical
No chance	0%
Almost no chance	$0 < p \leq 5\%$
Very unlikely	$5\% < p \leq 20\%$
Unlikely	$20\% < p \leq 45\%$
Roughly even chance	$45\% < p \leq 55\%$
Likely	$55\% < p \leq 80\%$
Very likely	$80\% < p \leq 95\%$
Almost certain	$95\% < p < 100\%$
Certain	100%

models (Cawson et al., 2020). However, BARD remains the first tool to apply Delphi elicitation to the entire BN building process.

2.4.3. Probability Elicitation

BN software uses exact probabilities for its underlying computations, but probability estimates—whether from data or beliefs—usually come with some recursive meta-uncertainty, i.e., "vagueness." Understandably, users can be uncomfortable specifying exact probabilities, even when informed that they need not be treated as precise. One alternative is to replace them by a small number of standardized verbal terms (Chris, 1987; van der Gaag, Renooij, Witteman, Aleman, & Taal, 1999), and interpret these as numerical intervals: an approach formally adopted by the intelligence community in the ICD-203 mapping (Table I), however, how people understand such verbal terms varies (e.g., Wintle et al., 2019). Another alternative is to allow users to indicate their degree of uncertainty by eliciting intervals directly, i.e., eliciting more than one point. Such protocols include a 3-pt method (Hanea et al., 2017; Malcolm, Roseboom, Clark, & Fazar, 1959) and a 4-pt method (Speirs-Bridge et al., 2010).

Several structured protocols have been developed and utilized for eliciting and aggregating judgments from groups of experts that can be applied to BN probabilities, whether single or multipoint: notably Cooke's (Colson & Cooke, 2018), SHELF (O'Hagan, 2019), and the Delphi-style IDEA (Hanea et al., 2017; Hemming, Burgman, Hanea, McBride, & Wintle, 2018). One study used another form of Delphi to elicit exact probabilities for CPTs from groups (Etminani, Naghibzadeh, & Peña, 2013).

2.4.4. Sources and Validation

During elicitation, it is an important but often underappreciated task to document how the model was constructed, e.g., the sources of modeling elements and their reliability. As yet, there is no accepted standard for this kind of meta-documentation.

Validating computer models means testing their accuracy in representing a real-world system. BNs are generally validated using the same kinds of information sources by which they are learned: data, expert feedback, or a combination of the two (Flores et al., 2011; Korb, Geard, & Dorin, 2013). This dovetails well with iterative development, since fresh, independent data or opinions can both validate and extend the current model. Furthermore, structured group deliberation such as Delphi incorporates a degree of validation for the group product, to the extent that individuals are persuaded to provide independent critiques rather than merely nodding along. Pitchforth and Mengersen (2013) have proposed a general framework for expert validation that is more systematic than *ad hoc* exploration of "what if" scenarios. However, although the BN software packages listed above provide explicit support for some forms of data-driven validation, they universally neglect structured protocols for expert validation.

2.5. Explaining BNs

The automatic generation of explanations from BNs was first investigated around 1990 (Boerlage, 1992; Sember & Zukerman, 1989; Suermondt, 1992), but has seen relatively few advancements over the subsequent three decades (Jitnah, Zukerman, McConachy, & George, 2000; Keppens, 2011; Korb, McConachy, & Zukerman, 1997; Vreeswijk, 2005; Zukerman, McConachy, & Korb, 1998; Zukerman, McConachy, Korb, & Pickett, 1999). Recently, there has been some renewed interest, e.g., in the "progressive" explanation of BN inference (Kyrimi & Marsh, 2016), and in "story-based" idioms for interpreting BNs (Vlek, Prakken, Renooij, & Verheij, 2016). Many explanatory algorithms have been special purpose, with language tailored to specific variables and subnetworks, and/or had little capacity to explain complex nonmonotonic relations. BARD's explanation-generation component (Section 3.3) is general purpose, using language applicable to any variables while exploiting common idioms for expressing probabilistic and causal relationships, and

includes novel features for explaining complex dependence relations.

2.6. Delphi Protocols for Group Decision Making

There is considerable evidence that decision making by groups (either by reaching consensus or amalgamation) can produce better outcomes than decision making by individuals (Charness & Sutter, 2012; Kugler, Kausel, & Kocher, 2012; Salerno, Bottoms, & Peter-Hagene, 2017; Straus, Parker, & Bruce, 2011). However, there are also well-known problems while working with groups, e.g., anchoring on the earliest responses, "groupthink," and the excessive influence of higher ranking members (Kahneman et al., 1982; Mumford, Blair, Dailey, Leritz, & Osburn, 2006; Packer, 2009; Stettinger, Felfernig, Leitner, & Reiterer, 2015). Several methods have been developed over the years that attempt to harness the positives of groups, while preempting or mitigating the negatives; one of the most well-established is the Delphi technique (Linstone & Turoff, 1975; Rowe, Wright, & Bolger, 1991).

Delphi is an example of a nominal group technique, where the group members never actually meet face-to-face, but interact remotely. Thus, participants need not be present at the same location or make their contributions synchronously, i.e., at the same time—practical benefits when participants are dispersed, perhaps internationally, with limited time and conflicting or busy diaries. Furthermore, participants do not even know who their fellow group members are—a deliberate ploy designed to ameliorate cues related to supposed seniority, experience or expertise, which may be unhelpful (since perceived expertise and advancement can often be related to personality or background characteristics, rather than skill or knowledge). Thus, members can focus on the information provided by others, and the undue influence that powerful or dogmatic individuals can have on group judgments is reduced.

These anonymous participants are first asked to provide their own judgment on the issue at hand, before finding out about the responses of others. This increases the independence and diversity of initial responses, reducing "social loafing" and the premature conformity seen in anchoring and groupthink. The responses are collated by a facilitator, then fed back to the participants for a second round. The participants consider the information (which may simply be the mean or median of the group response when quantitative values are in question, but may also include rationales/justifications for answers), then provide another response, which can either be unaltered or amended. This encourages participants to rationally reconsider their response in the light of any new information provided by others.

Several rounds may take place, continuing until some stability is achieved (i.e., opinions no longer change significantly)—although most changes take place in the second round, and few studies go beyond two or three rounds. This process tends to increase the level of consensus in the group, but the more fundamental aim is to increase the overall quality of the responses, e.g., improve the mean accuracy of estimates relative to some ground truth rather than merely decrease the variation between estimates. After the final round, the facilitator usually aggregates the responses of the individual members (or collates them, if responses are qualitative in nature), and the resultant answer is taken as the group response. Answers are usually weighted equally, which ensures that the final response reflects fairly the views of all group members. In addition to their benefits for administration and collation, using a facilitator tends to encourage constructive contributions from members and avoid any unproductive, heated arguments.

We shall call a process *traditional Delphi* if it includes all the features just described. However, there have been many variants that omit some features, yet, by family resemblance, have reasonably been called Delphi processes, despite the objections of purists. To be clear, we will use a slightly more general definition of *Delphi* that includes traditional Delphi as a specific variant. Following Rowe et al. (1991), the necessary and sufficient criteria are *anonymity*, *iteration* followed by *feedback*, and *aggregation* of group responses.

One Delphi variant, *Real-time (RT) Delphi* (Gordon & Pease, 2006), is not only asynchronous within rounds, but entirely "roundless": the iterative process (providing individual responses, viewing information from other participants, and amending responses) is not synchronized by a facilitator even at the end of a round; rather, each participant can iterate immediately, even if other participants have not yet done so. This is even more flexible in the timing of participants' contributions, and can potentially speed up the Delphi process. However, since participants can see any other available responses directly and asynchronously, rather than after amalgamation of every member's response by a facilitator, some of the biases

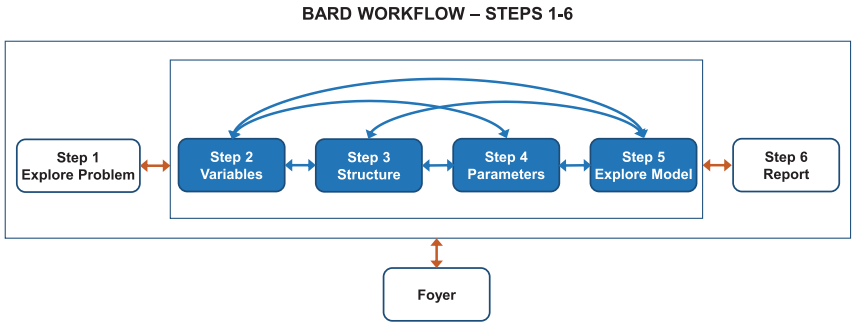


Fig 2. The BARD workflow consists of six steps. From the Foyer, users choose which of their problems to work on. Analysts and the facilitator can then move flexibly backwards and forwards between steps to update their work as desired, with BN modeling occurring in Steps 2–5.

associated with direct interaction may reemerge. The social process in BARD is a version of RT Delphi; we discuss our reasons for trading off speed and ease of use against bias in Section 3.2.

3. BARD APPROACH

BARD supports a highly structured approach to eliciting and building BNs, which is stepwise, incremental, and iterative. Furthermore, in Delphi-style, individual group members (called *analysts* in BARD) submit their initial contributions to a problem blindly, and all contributions anonymously. A moderator (called the *facilitator* in BARD) usually guides and supports the analysts through the process, and ensures an overall group solution is produced. Section 3.1 provides an overview of the workflow, and Sections 3.2–3.5 provide further details on specific aspects.

3.1. BARD Workflow

The BARD workflow consists of six steps, as depicted in Fig. 2. The premodeling, preparatory Step 1 focuses on helping the group understand the problem to be solved and the questions to be answered, along with the main hypotheses and pieces of evidence. In Steps 2–5, the focus is on building a causal BN that models the problem situation and using the causal BN's reasoning to assist in answering the questions. These steps reflect the natural sequence of tasks in BN construction: selecting the variables (Step 2), determining the network structure (Step 3), parameterizing the model by eliciting the CPTs (Step 4), and then exploring the completed model's reasoning on specific scenarios (Step 5). Step 6 focuses on producing a structured written report. While there is a natural sequence to the workflow, it is not a one-way street: users can always go back to revise their previous work. This supports building the BN iteratively and incrementally, in accordance with best practice (Section 2.4).

Analysts contribute their individual domain knowledge and problem-solving abilities across these six steps, while the facilitator constructs a group version for each step based on the analysts' work. This is done via a structured workflow *within* each BARD step. At each step, analysts are required to first work on their own, and then share that initial attempt with the group. After this, they can view other analysts' work and the current group solution (Fig. 3), discuss solutions via the step-specific discussion forum, and move on to the next step whenever they choose. Analysts can also move back to an earlier step to revise their work at any time, and then move forward to any step they previously reached.

The facilitator's workflow is more flexible than the analysts', as they can move to any BARD step and view all analysts' shared work at any time. In addition to encouraging constructive analyst engagement, the facilitator is tasked with copying what appears to be the best solution at each step to the group workspace, possibly by amalgamating more than one analyst's work. Optionally, the facilitator may announce deadlines at which they intend to progress to specifying the group solution for the next step. If no clear consensus has yet emerged on a single best solution, then it may require some judgment—or a request for supplementary analyst input—to identify which solution or consistent set of components to adopt. The facilitator can present the current group solution (for any or all steps) back to the group at any time so that analysts who have shared their work for that step can view it, discuss and provide feedback, and revise their work if they wish. If an analyst modifies an earlier answer after the facilitator has already amalgamated the responses for that step, then the facilitator can, if they judge it worthwhile, incorporate such a change and/or ask other analysts for their

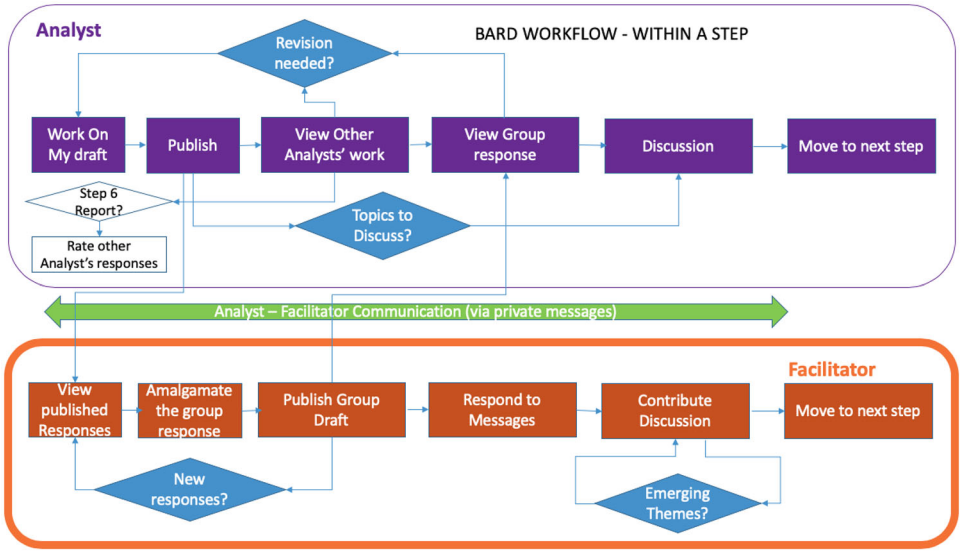


Fig 3. High-level representation of the BARD workflow within a step for analysts (above) and facilitator (below) working in a BARD group.

opinion.

If a minority of analysts favor a different model to the majority, then there is no software impediment to developing this in their own workspaces, conferring between themselves, and drafting an associated minority report; whether this should be encouraged by the facilitator depends on the context. In the absence of a facilitator or a facilitator decision, analysts may continue to work independently on their solutions, and the platform provides a rating mechanism for analysts to select among themselves the completed model and report they think is best. Thus, BARD is designed to benefit from facilitation and consensus where available, yet not be overreliant on facilitator expertise or convergence of opinion, and allow solutions to be produced even without them.

In Delphi processes, one possible stopping rule is to conduct further rounds until there is no significant change in opinions (which may still be diverse). While this may be used in BARD, it is not the default: completion of the workflow was deadline-driven in our experiments, and this is likely to be more applicable to future real-world applications.

3.2. BARD Social Process

3.2.1. Delphi Implementation

The social process within BARD's structured workflow is a flexible variant of RT Delphi, arising from our observations during our interactive design and prototyping. It reflects both the demands of the task and the needs of participants. BN building and reporting is a much more complex and demanding task than typical Delphi applications, so the BARD process would be far too drawn-out if multiple Delphi rounds were employed for each of the six steps, or if all analysts were required to return to a previous step whenever one wished to iteratively revise their work. The problem of coordinating analyst input was exacerbated in the CREATE program where participants were working asynchronously, and many worked in a few bursts of activity within the problem-solving period rather than providing more continuous input. We found that participants expected significant autonomy, e.g., to be allowed to complete and comment on steps others had already passed through and/or not yet reached. They often preferred direct engagement with their peers, rather than through a facilitator; and since the facilitator sometimes absented themselves temporarily or permanently, it was important to allow groups to progress and complete the task without one.[12]

[12]During BARD's development, we prototyped and evaluated other versions of Delphi where analysts are more constrained, two of which are still supported and configurable via the administration panel at problem setup time. These are (1) like the default version, except that analysts all move to the next step at the same time when the facilitator gives them access, and (2) a very traditional version, where there are multiple Delphi rounds for each of the six steps of BARD, the facilitator controls ac-

We found that many participants appreciated one of the key aspects of our Delphi implementation: not only is each analyst asked to make an independent initial attempt to answer each question, they can retain and develop their own model throughout the process. However, for complex tasks, such duplication of effort can become a disadvantage compared to a more distributed approach. BARD mitigates this by providing functionality enabling users to select and automatically incorporate elements of each other's work into their own solution (if they are analysts) or into the group solution (if they are the facilitator). This is done in slightly different ways in each step, due to the distinct types of content being incorporated. For example, if one analyst likes some of the variables another analyst has defined in Step 2, then they can easily copy selected variables and proceed to demonstrate a slightly different structure in Step 3. Thus, as well as greatly reducing the burden on individual analysts, copying also makes it easier for them to to make compatible contributions.

We also found that participants varied in their domain expertise or problem-solving ability. Hence, although facilitators may often adopt the most prevalent opinion or an equal weighting of parameter estimates, the software does not rigidly enforce it—allowing them to take into account group discussion and adopt what they judge to be better answers.

Although analysts' real names are concealed, they are assigned pseudonyms that they keep throughout the problem. This helps to identify the work and comments of each analyst in each step, and also relate it to their work and comments in other steps, which makes discussion and comparison far easier for participants.

Finally, throughout the six steps, BARD encourages its users to enter a rationale to explain their analysis, making it easier for other group members to understand the solution. This detailed documentation improves the exchange of ideas and provides a basis for discussion on points of disagreement, hopefully leading to a better understanding of the problem and the resulting solution.

cess to the next step, there are no discussion forums, and analysts only see the amalgamated group model provided by the facilitator rather than each other's work directly.

3.2.2. Social Roles

A BARD group consists of a single *facilitator* and multiple *analysts*. The analysts contribute their individual domain knowledge and problem solving abilities across the six steps of BARD and are tasked with producing the best possible solution to the problem the group has been given. The number of analysts is not limited by the software, but was six to nine in our experiments, where we aimed for a large number of groups, yet each with substantial social interaction despite some inactivity or attrition.[13]

Although facilitators can be very useful, it can be risky to rely heavily on either their expertise or activity. BARD is very flexible about their contribution, which can vary considerably depending on the context and the individual. At one extreme, BARD can be deployed with a BN expert as the facilitator, as in more traditional BN elicitation. At the other extreme, BARD can be deployed without a facilitator at all, e.g., where human resources are very limited. Our default approach, however, which reflects CREATE's aims, is embodied in BARD training, and was validated in our initial experiments, is to use a volunteer facilitator who has no more prior expertise than the analysts.

After being given the analyst BN training, interested participants may opt to receive a little additional training in facilitation. Their role is primarily to act as a traditional Delphi facilitator, i.e., promote constructive engagement and the development of better solutions, and, ideally, a rational consensus. For this purpose, the facilitator may provide basic instructions, signal when responses are to be submitted, monitor activity levels, encourage discussion of emerging points of difference, discourage antisocial behavior, and present amalgamated results back to the group.

Amalgamation, here, means selecting or constructing the group solution at each step. The facilitator is trained to respect any consensus and incorporate good contributions from multiple analysts where possible, while avoiding making unnecessary novel

[13] At the lower end, Delphi is often performed with groups as small as five active members, but even two analysts could find BARD useful to build and compare models. At the upper end, it may well become onerous for each analyst to review more than 20 peer models. However, in some crowdsourcing applications there may be high attrition among analysts, or many analysts who prefer (and are permitted) to comment on others' models rather than build their own. So, if groups of 100 analysts tend to produce only 10 models between them, then this could be an effective configuration.

contributions themselves. For the final BNs and reports, the facilitator is also assisted by the ratings provided by analysts. In our experiments, we received no complaints from analysts about dictatorial behavior from facilitators. However, the facilitator is entrusted with the final decision on what is included and editorial control over how it is expressed in the group's final report.

The BARD platform supports a subsidiary role: an *observer*, who is assigned to a group and can observe all stages of their BARD process in "view-only" mode, i.e., without being able to contribute to it (apart from messaging the facilitator). They are able to see all public contributions from their group members and facilitator, and all steps at any time. This was originally introduced for the CREATE testing program, to support reserve participants ready to step in as replacements if other participants dropped out of a group during the problem-solving process. However, the observer role has also proved useful for researchers to monitor participant progress during experiments, for educators to monitor student progress during teaching applications, and to allow the analysts and facilitators themselves to retain view-only access to their problem after it "closes."

A BARD *administrator* has ultimate control behind the scenes over how the platform is configured and run. Their special technical capabilities are listed in Section 3.4.

3.2.3. Communication

BARD advocates and supports participants using pseudonyms to maintain their anonymity, which is a characteristic Delphi feature. However, this aspect is managed by the BARD administrator (per Section 3.4) who may choose to have users identified by real names rather than pseudonyms if this is deemed more appropriate.

BARD provides two main communication channels: discussion forums and messaging.

Discussion Forums. There is a separate discussion forum for each BARD step, where group members who have published an initial answer for this step can communicate directly with each other about it. By discussing any difficulties, providing feedback on others' work, and gaining a better understanding of the reasoning behind it, members can increase the chances that improvements will be adopted and/or a consensus reached. A new discussion on a particular topic can be started by any analyst or the facilitator. Each topic's discussion is displayed as a single thread, with participants encouraged in the BARD training to use the `@analyst_pseudonym` convention to indicate when their comment is a reply to another analyst's comment.

The provision of a separate discussion forum for each step is intended to support the Delphi principle that participants should attempt their own solution before viewing other analysts' contributions; for example, an analyst may be able to read and contribute to a discussion about the BN variables (Step 2), but if they have not yet provided their attempt at the BN structure (Step 3), then they cannot see the Step 3 discussion forum. Of course, this relies on the analysts following the protocol and not discussing topics in one forum that are related to a different step.

Messaging. BARD's chat message channels provide private, two-way messaging between the facilitator and an analyst, and BARD also allows the facilitator to send a single message to multiple analysts. Messages are not associated with steps. Analysts do not see who else the facilitator may have sent the same message to, and they do not see any messages between the facilitator and other analysts. The message sender may optionally elect to generate an email notification for the recipient(s), which is a useful nudge for someone to login again to BARD when it is being used by the group asynchronously. However, BARD does not allow direct messaging between analysts, to reduce private "side" conversations[14] and to encourage a collaborative process where all group members have access to the same information and discussions.

BARD training advocates that the majority of facilitator communication to group members should be via the discussion forums. However, the messaging channel is more suitable for (1) contacting individual analysts who have not been contributing either their own answers or to group discussion; (2) answering private questions from an analyst about using the BARD application, especially where to find help; and (3) communicating privately to an analyst regarding inappropriate social behavior, such as showing a lack of respect for others in the forums. The facilitator is supported in these aspects of the role with an

[14]It is difficult to remove all chances of side conversations while enabling public discussion, because analysts could, for example, post their private email addresses onto the forum and set up a conversation outside of BARD.

administration panel that shows a summary of each group member's last BARD access, and the step they have reached.

3.3. The Six Steps of BARD

Here each of the six steps (Fig. 2) are described in more detail.

3.3.1. Foyer

To commence or recommence working on a problem, users first log in to their account and arrive at the *Foyer* page. Here they see a list of all problems to which they have access, and can choose which problem workspace to enter. They will also see any messages for them, including activity updates on any of their problems that are still "open" for contributions, such as which other group members have been active since they last logged out.

3.3.2. Step 1: Explore Problem

This allows analysts to read and examine the problem, review any questions that have been posted, and encourages them to extract key features by identifying (1) the hypotheses suggested by the problem/questions and (2) the items of evidence most pertinent to those hypotheses. Analysts are also encouraged to provide rationales for the inclusion of each hypothesis and evidence item. The motivation behind this step, as a precursor to the BN modeling, is to have the group gain and record a shared understanding of the problem they must solve, which reasoning guides (e.g., Clapper, 2015) suggest is crucial to clarify upfront, and reach some level of agreement on the key elements that must be included or addressed in BN construction.

3.3.3. Step 2: Variables

Here is where the variables of the BN are specified, with BARD suggesting that analysts consider converting the hypotheses and evidence items from Step 1 into variables. However, the only special type of variable BARD asks users to identify here is "target" variables, with the remainder classified as "other" variables.[15]

[15]The option to use more diverse labeling for variables is a feature we may add to BARD in future, e.g., to help identify critical assumptions unsupported by quantitative evidence.

Target variables are those most closely associated with the questions to be answered. Targets are often the identified hypotheses, but this is not always so: e.g., if the problem is to estimate the probabilities of specific symptoms occurring, then the targets are these symptoms, whereas the possible diseases may have been identified as the causal hypotheses (for which there may be various kinds of evidence). BN modeling methodologies also describe targets as "query" or "output" variables, and suggest identifying them first before focusing on other variables that are either their causes or effects.

Other variables here may include quantitative evidence, relevant background beliefs that are not naturally described as evidence, and intermediate variables that link either of them to the targets. Target variables are distinguished with a different color on the graph visualization, which is displayed in the subsequent BN building Steps 3–5. However, any variable may be designated as an output variable of a "scenario" in Step 5 (see below).

Variables must be specified together with their discrete states. BARD currently supports four categories: (1) Boolean for propositional variables, i.e., with just the two states *True* and *False*, e.g., *Testing performed* in the Drug Cheat example; (2) Binary, i.e., any other two-state variables, e.g., *Taking M897*; (3) Ordered, i.e., any multistate variables with the states in ranked orders, e.g., {*High, Medium, Low*}; and (4) Unordered, i.e., any other discrete variable, e.g., the *Event* variable has the states {*Weightlifting, Running, Swimming*}.

Descriptions of the variables and the variable states are solicited, but not required, as are "rationales" for the choice of variables. These meta-data items are intended not only to document the intent and meaning being these modeling elements, but also to stimulate active discussion when other analysts see them and disagree.

3.3.4. Step 3: Structure

This is where the relationships between the variables are specified. In this step, BARD displays each variable as a draggable, named node on a canvas, and prompts analysts to specify the causal structure by drawing arrows between pairs of variables, i.e., graphically specifying directed links between nodes to produce a "network" diagram for the BN. Target variables are differentiated from other variables by color (Fig. 1). Arrows can be readily deleted or redirected to a new variable. Analysts can associate text

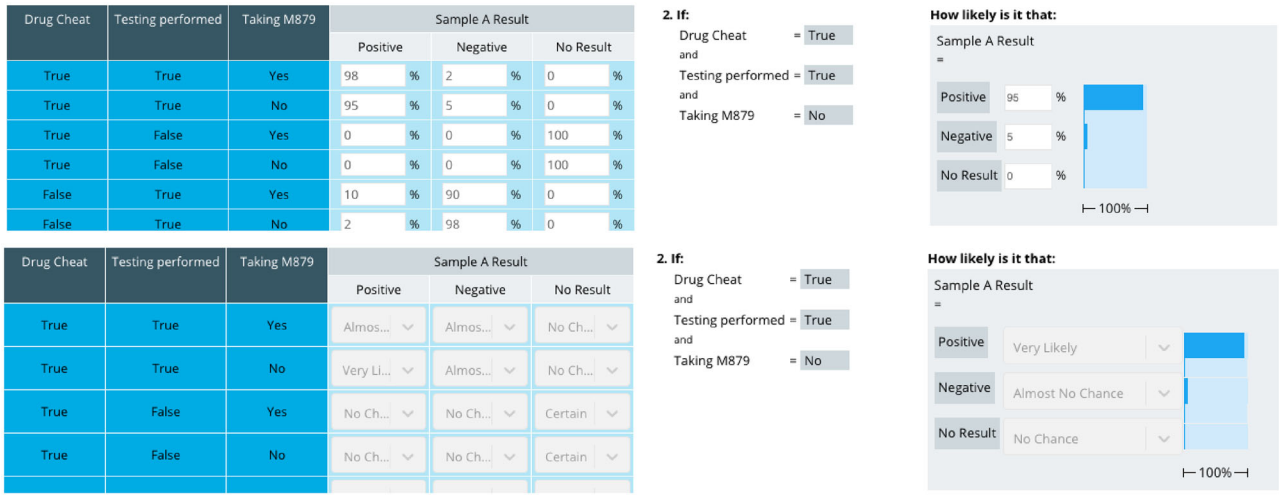


Fig 4. Examples of the four input modes available in Step 4 Parameters: percentages above, qualitative descriptors below; table left and question-based right.

labels with arrows, as well as create general labels anywhere on the canvas to act as titles or general purpose on-canvas documentation (a standard feature in most BN software GUIs).

During this step and later steps that display a network view of the BN (e.g., Step 5, as shown in Fig. 5), BARD adjusts the layout of the network to enforce a natural causal "flow" (left-to-right and up-to-down) and prevent graphical elements from overlapping. This is achieved using a technique called *constraint-based layout (CoLa)* (Dwyer, Marriott, & Wybrow, 2009).[16] As the analyst specifies arrows, CoLa automatically shifts variables around the canvas to maintain distances between them, while preventing variable overlaps and minimizing overlaps between arrows and labels. One advantage of this automation is reduced effort by the user, but another is that it is easier for users to recognize similarities and differences in other models when they are laid out in the same way. CoLa does allow analysts to reorganize the network layout manually, by clicking and dragging variables around the canvas, while still enforcing some layout constraints.

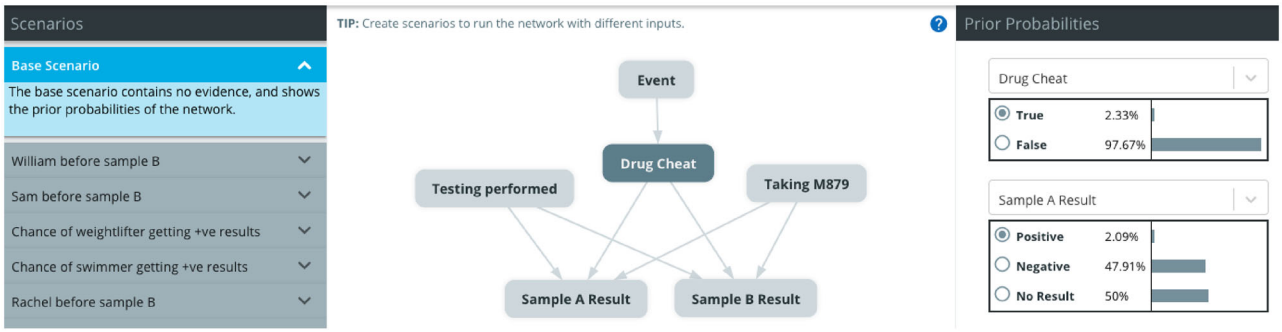
At this point, the analyst may wish to add additional variables or modify the states or names of existing variables. They can do by returning to Step 2, per the flexible BARD workflow across steps (Fig. 2). When completed, the influence diagram produced in Step 3 usually already helps to clarify what factors are relevant and interconnected, even before specifying the quantitative nature of these connections in the next step. If time is pressing, then the analyst or group can proceed immediately to report writing in Step 6.

3.3.5. Step 4: Parameters

This step allows users to specify the conditional probabilities for each child variable given each joint state of its parents, i.e., the child's CPT. BARD provides two modes for specifying the conditional probabilities: (1) as answers to questions, with one question for each combination of the parent node states; or (2) via a table.[17] In either case, BARD provides two ways of entering probabilities: (i) as percentages; or (ii) as English language verbal descriptors, each of which has an associated probability range, as specified in ICD-203 (Clapper, 2015) and shown in Table I. In combination, this yields four possible input modes, as depicted in Fig. 4. This approach caters both for users who prefer to model with precise parameters and for users who prefer to avoid the appearance of "false precision" by using qualitative

[16] Currently, the configuration of these layout constraints is done in the software. In future, we plan to make this configurable by the BARD administrator or by users.

[17] We used a table layout similar to Netica rather than Hugin, which reverses the rows and columns, because this makes scrolling vertical rather than horizontal when the table size increases.



(a) The base scenario (expanded on left), with five more scenarios listed (but collapsed), each including different sets of evidence.

(b) The third scenario listed, which includes the three pieces of evidence available before Sam's Sample B result is known.

Fig 5. BARD screenshots for the Drug Cheat Problem at Step 5: Explore Network. Evidence can be added into scenarios in the left panel, and updated probabilities for the chosen output variables are shown in the right panel. The network structure is shown in the center panel with the the evidence variables highlighted in blue, and below this is a summary verbal explanation.

verbal descriptors (per Section 2.4). Behind the scenes, BARD does not substitute for each verbal descriptor the endpoints of the associated numerical intervals and perform extreme-case computations, which quickly become very imprecise. Rather, BARD substitutes a central point, then preserves the ratio of these points while normalizing the distribution, e.g., if all states are assessed as "highly probable," then the normalized distribution is uniform. This is a technique previously used in other BN tools, e.g., Nicholson et al. (2011).

Step 4 now also provides some support for learning the CPTs from data, in the form of Netica's simple "counting-learning" functionality.[18]

Where individual probabilities have not been specified, BARD warns the user of that fact, but allows the user to proceed and defaults to a uniform distribution of any unused probability mass. This

[18]This functionality was implemented after the IARPA CREATE program concluded, so it was not available to users in the experiments reported in Section 4.

allows quick specification of CPTs for which only a proper subset of the parameters are known.

3.3.6. Step 5: Explore Network

This is where the group members can use the BN for reasoning, thus exploring the consequences of Steps 2–4. Evidence is added by setting one or more variables to particular states, and the BN reasoning engine computes new probability distributions for the remaining variables. In BARD, each set of evidence is called a *scenario* (following AgenaRisk terminology), and may involve setting values for any number of variables. A scenario may describe a specific situation given in the problem description or just a hypothetical "what-if" scenario that the analyst wants to explore. In BARD, scenarios can be saved, named, given associated descriptions, shared, and discussed. When viewing another user's BN, a second BARD user cannot edit it, but they can explore its consequences by adding new scenarios that are only visible to the second user. Analysts can always alter or extend their own BN or the associated scenarios, and facilitators can do this for the group's model. Step 5 always includes a default "base" scenario, which shows the probability distributions for all specified output variables when no evidence has yet been added.

Scenarios allow an explicit and visual way of investigating whether the BN gives a reasonable representation of known or hypothetical situations. They also provide a direct means of answering questions about the confirmatory value of evidence or the final probability of some event given any combination of evidence. More formally, by allowing scenarios to be set up, stored and examined, BARD allows analysts to undertake the following validation activities (Korb et al., 2013): *face validity*, i.e., checking whether a model captures the known features of a situation; *content validity*, i.e., checking whether the model's confirmatory or causal relationships capture known relations; *case analysis*, i.e., seeing whether known cases are modeled correctly; and *sensitivity analysis*, i.e., determining whether variations in target variables are proportionate to variations in evidence, including examining the confirmatory power of different evidence sets. In the future, we anticipate providing more targeted sensitivity analysis tools, such as reporting Bayes Factors (for confirmation) or causal power (e.g., via the measure in Korb, Nyberg, & Hope, 2011).

The BARD Step 5 workspace is divided into three panels (Fig. 5): The left panel contains the scenarios, with the base scenario listed first, and a single active scenario (selected and expanded) at a time; the middle panel shows the BN structure; the right panel shows the output variables (any subset of the variables, as selected by the user) together with the computed probability distribution over their states; and a summary explanation is shown below the middle panel. Thus, when a scenario is active, the altered distribution over nonevidence nodes may be easily examined. If the user wants to compare the outputs of several scenarios, they can either click back and forth in the left panel (like AgenaRisk) or instantly open duplicate instances of BARD in new browser windows to view the scenarios side-by-side.

BARD Step 5 includes a general-purpose AET, implementing a mix of traditional and novel natural language generation techniques, and taking advantage of the explicitly causal nature of the links and common idioms for expressing probabilistic and causal relationships. This can be used by analysts either to critique the BN or to contribute to writing up a report.

The AET generates both a summary explanation on the Step 5 tab, and a detailed explanation accessible in a separate dialog box. For both, target variables and states specified on the Step 5 tab focus the explanation. Probabilities are stated both numerically and with verbal descriptors, following the ICD-203 recommendations (Table I). The summary explanation states the target probabilities if no evidence were entered, the evidence specified in the scenario, and how the target probabilities change given this evidence (e.g., in Fig. 5(b)). The detailed explanation also includes the causal structure of the model, how the target probabilities are logically related, the general reliability and bias of the evidence sources, why the evidence sources are structurally relevant, and multiple ways to express the probabilistic impact of the individual evidence items on the targets that note any major interactions (Zukerman et al., 2019).

The AET has been tested on the 10 BARD training problems (Section 3.5), the four problems used in our human experimentation (Section 4), and three additional problems developed for CREATE. Overall, it has been shown to produce satisfactory English language descriptions. However, we have yet to experimentally test to what extent the provision of these automated explanations, or which elements of them, improve analytic solutions.

When the group has explored the consequences of the BN and is finally satisfied with it, they can move on from the spiral prototyping of Steps 2–5 to writing their report in Step 6.

3.3.7. Step 6: Report

This step provides an environment in which the group can develop, in the same Delphi style, a joint written answer to the questions raised in the problem statement. In our preliminary testing, we found that providing a template to assist users in writing effective reports improved their performance. The template encourages analysts to methodically organize and explain their analysis in detail, and prompts them to include various important elements of good reasoning, such as key assumptions and probability estimates for their conclusions (Clapper, 2015). The AET in Step 5 also provides its detailed output in sections of text aligned with the template sections, which can be used verbatim or paraphrased to complete corresponding sections of the template.[19]

Step 6 also allows analysts to rate final reports, which will either decide or inform which report is selected as the solution for that problem. This element was added when preliminary usage indicated that discussion forums did not always generate a clear consensus or sufficient guidance for the facilitator on the best BN or final report. Furthermore, in some experiments a few facilitators were no longer active at this stage in the process, so analyst ratings allowed the group report to be selected automatically in these cases. Rating is done on a scale from 1 to 10 using a slider. After submitting their rating, an analyst can see the average score for each option and how many ratings have been submitted so far, but cannot see other analysts' ratings individually. If there are two incompatible but highly rated models or reports, then the facilitator could choose to collate them as two distinct, alternative views in the group's report, e.g., of the majority and minority.[20]

BARD includes functionality to officially "submit" the final group solution, which is useful for human experiments and/or applications where there is a hard deadline. In Step 6, the facilitator can voluntarily click a submit button, which generates a PDF of the group report and sends it to a nominated electronic location. If facilitator submission does not occur by a specified deadline, then automatic submission can be enforced according to configurable rules, e.g., selecting the highest rated report. All group members have access to the final published group solution, whether it has been submitted or not, and can download both the BN (in Netica or AgenaRisk format) and the written report (in PDF format).

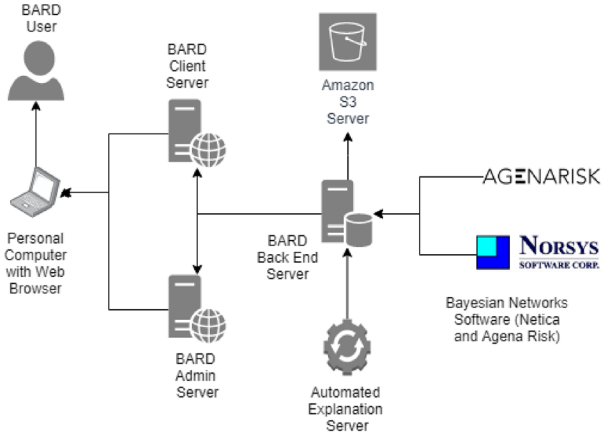


Fig 6. BARD application architecture.

3.4. The BARD Platform

BARD is a client-server application comprised of a group of cloud-based servers that provide services and resources to connected clients (Fig. 6). The main BARD server provides the BARD login, collaboration, problem solving and report generation services, and connects: (1) a Database server, which provides SQL database services; (2) a BN server, which provides the back-end reasoning via commercial BN software[21]; (3) an Automated Explanation server running the AET (Section 3.3, Step 5), which also utilizes the BN server; and (4) a Storage server,[22] which stores items such as images uploaded into discussion forums or the report. BARD is currently hosted on Monash University servers, to which free access is available on request for research purposes,

[19] For an example of an model solution written within (and displaying) BARD's integrated report template, which was annotated and presented to experimental participants in Korb et al. (2020) as part of their training, see "Smoking and Cancer - Problem Statement" at https://bit.ly/2V2rwl4 and "Smoking and Cancer - Annotated Solution" at https://bit.ly/3icbBKd.

[20] Such rival models could also be exported for prediction via model averaging, which is a common Bayesian technique but not directly supported in BARD.

[21] BARD can currently use either a Netica or AgenaRisk BN server; we anticipate extending this to other widely used BN software.

[22] BARD currently uses the Amazon Simple Storage Service (S3).

and under negotiable terms for commercial purposes. For any users who need a more secure environment, BARD software can be made available for users to install on their own servers. This was a necessary feature for our U.S. intelligence clients, and occurred during the CREATE experiment run by IARPA.

The BARD platform has an administrator console, which allows the BARD administrator to modify some software behavior: configure the application process flow, enable/disable certain features (e.g., the version of Delphi), and schedule tasks. The console also allows the administrator to manage problems, users, and groups, including: set up problems, optionally with start and end times; create user accounts; create groups to work on a problem; allocate users to groups, optionally with pseudonyms; allocate roles to group members; upload a BN (from Netica or AgenaRisk format) to any user's workspace to demonstrate a partial, previous, or ideal solution; and download the BNs and reports users have produced.

While BARD has been developed as a collaborative BN tool to improve analytic reasoning, stripping out the collaboration features still leaves a sophisticated tool for an individual to analyze and solve problems. We call this version *SoloBARD*. It allows an analyst to move through the six steps of BARD without consultation or guidance from anyone else.

3.5. BARD Training

The BARD platform comes with approximately four hours of training, covering all key elements of the BARD approach: (1) causal BN technology; (2) the BARD workflow including the six steps of BARD and the group interactions; (3) the BARD software tool, from the perspective of both analysts and the facilitator; and (4) writing structured analytical reports, using the BARD templates.

This training consists of interactive e-courses for individuals, produced using the StoryLine 360 tool and hosted on a commercial cloud-based Learning Management System (LMS), called Moodle. The e-courses are all relatively short (2–15 minutes) and discrete, which suits self-paced learning. The LMS allows the BARD training material to be repackaged into different courses for specific purposes, and partitioned as desired. For example, in one experiment with BARD (Section 4) we presented the training courses to participants in the partitions: "required" (approx. 1 hour 30 minutes), "recommended" (approx. 1 hour 15 minutes), and "optional" (approx. 1 hour 15 minutes).

Although users are asked to do some of this training upfront, the LMS is integrated with the BARD platform, so that users can revisit the LMS at any time from the BARD landing page. Integration also allows training activity and completion data to be exchanged with BARD, which can then be used to guide group creation and role allocation.

These LMS-hosted e-courses are augmented by several other forms of help and training material within the BARD tool, which provide further assistance and guidance for users in real time as they work through the BARD process. These embedded components include: *training problems* that allow users to work through elements in the BARD approach as an individual analyst, with an "ideal" solution progressively revealed at each step as the prepopulated group solution; an optional *product tour* associated with each BARD page, offered on first use and remaining available later; a *general Help facility* that includes both the e-courses and PDF versions of them; *context-specific help* as tooltips, page-based tips, and pop-up help tips; and *"What do I do next?"* guidance.

4. EVALUATION

Here we summarize two experimental studies to test the effectiveness of the BARD approach to problem solving, and their findings; each is published and reported in detail elsewhere.

4.1. SoloBARD (Cruz et al., 2020)

This experiment addressed whether the BARD system improves individual reasoning on probabilistic reasoning problems, using an independent measures (between-groups) design. In the experimental condition ($N = 29$), individuals received selected BARD training and used the SoloBARD system—which provides the six steps of BARD without any of the social processes—to construct BNs and produce written solutions to the problems. In the control condition ($N = 30$), individuals received generic training based on the CREATE "Guide to Good Reasoning" slides[23] and produced their written solution using Microsoft Office tools.

Our cognitive psychologists developed and tested the three problems used, and their problem-specific marking rubrics. Each problem appeared simple, and could be solved precisely with a simple BN of only five to eight binary variables with zero

[23] Available at https://bit.ly/2WJtpQJ.

to two parents each, but nevertheless incorporated a known, major reasoning difficulty (Liefgreen et al., 2018; Pilditch, Hahn, & Lagnado, 2018; Pilditch et al., 2019). Participants were asked *explicit questions* that required some explicitly specified information, either qualitative (e.g., "Which is the most likely hypothesis?") or quantitative (e.g., "How likely is it?"). They were also asked *implicit questions*: to give reasons for their answers, where the relevant observations were not explicitly presented to participants, but were listed in the marking rubric. One point was awarded for each answer or observation that participants fully included, and a half point for each observation only partially included.[24] Unlike subjective, global judgments about reasoning quality, these very specific rubrics are easy to interpret and apply. Participant answers were assessed blindly by external raters, who were trained to adhere closely to the rubrics, ignoring any extraneous material. Reasoning performance was summarized by two measures: (1) total score on both explicit and implicit questions; (2) total score on only the explicit questions. On both measures, the experimental condition performed significantly better than the control, with large and very large effect sizes respectively (Glass' Δ 0.8 and 1.6).[25] These results demonstrate that BARD, even when used privately by individual analysts, assists them in producing better reasoned reports for suitable problems.

4.2. Teams Using BARD (Korb et al., 2020)

This independent measures experiment addressed whether, given similar probabilistic reasoning problems to the previous experiment, teams using BARD submit better reports than control individuals using the best available pen-and-paper tools for probabilistic reasoning. In the experimental condition ($N = 198$), 25 teams consisting of six to nine analysts and a facilitator received BARD training and used the BARD workflow. In the control condition ($N = 58$), individuals received generic training from the "Guide to Good Reasoning," and specific training in probability calculation using both frequency format chain event graphs (see Gigerenzer & Hoffrage, 1995) and the elementary probability calculus, and used Google's online G Suite tools[26] to produce their solutions. (Our sponsors, IARPA, viewed individuals as the most ecologically valid control, since their primary aim was to develop possible alternatives to "business as usual" for intelligence analysts, who typically work alone with no special analytic tools. We could not afford a third condition where BARD-sized teams received control-style tools, due to resource constraints and attrition concerns.)

Participants were asked to solve a simpler problem in Week 1, and the next two problems—taken from the previous experiment and subdivided into two successive parts—in Weeks 2 and 3 and Weeks 4 and 5, respectively. This allowed us to investigate the value of BARD for dynamic problems, where evidence is updated and an initial analysis must be revised. External raters were again recruited to blindly assess the solutions against the problem-specific marking rubrics. The experimental condition significantly outperformed the control on each problem, with very large to huge effects (Glass' Δ 1.4–2.2), greatly exceeding CREATE's initial target. These results demonstrate that BARD groups can also beat individuals in producing better reasoned reports for suitable probabilistic problems, even when the individuals are using the best available pen-and-paper tools. Our increased effect sizes also suggest that BARD's Delphi-style collaboration combined beneficially and cumulatively with its other features—although implementation differences decreased absolute performances in both conditions compared to Cruz et al. (2020), so an interaction effect may have contributed.[27]

In addition, participants from the experimental condition were surveyed at the end of the experiment to capture feedback on BARD usability, us-

[24] For a simple example of this type of problem and rubric, which was used in Korb et al. (2020), see "Smoking and Cancer - Problem Statement" at https://bit.ly/2V2rwl4, and "Smoking and Cancer - Rubric" at https://bit.ly/2v92emU.

[25] Glass' Δ measures effect size in standard deviations of the control group, which is more appropriate than using Cohen's *d* with pooled standard deviation of the control and treatment groups when comparing the results of separate experiments on each of several possible treatments, as occurred in CREATE.

[26] These include a word processor and a spreadsheet for numerical calculations. See https://workspace.google.com/

[27] We summarize the evidence on this issue in Korb et al. (2020). This includes a follow-up experiment we conducted (Bolger et al., 2020) to verify whether, and explore how, BARD's Delphi processes contribute, focusing on the core BN-modeling task of deciding on causal structure. Accuracy of causal structure was measured in a simple, standard way by the *edit distance* (i.e., the number of arrows that differ) between the structure participants produced and the normatively correct structure. We confirmed that even minimal feedback from peers tended to improve individual answers, and found some interesting patterns in the errors analysts made that could guide both further BN training and the amalgamation of individual structural responses.

ing the System Usability Scale (Brooke, 1996) to give subjective usability ratings, and an open-ended questionnaire. Both the ratings and open-ended comments showed overall positive user satisfaction with the BARD software, although we note that the results were undoubtedly skewed in the positive direction because participants who dropped out of the experiment didn't complete the survey.[28]

In our training and these initial experiments with novices, we kept the modeling problems simple with answers that were easy to explain and evaluate, by providing sufficient information for there to be normatively correct causal structures and point probabilities. However, for real-world problems, the available information analysts must use to model the situation is often a diverse mixture, ranging from proven numerical results to contentious or vague subjective opinions. BARD includes appropriate functionality (e.g., Step 1 source analysis, Step 4 verbal estimates, Delphi-style discussion), but such problems await further experimentation.

5. CONCLUSIONS AND FUTURE WORK

We have presented a novel structured technique for collaborative reasoning and problem-solving that combines a logical procedure for building causal BNs with a Delphi-style social process. BARD is the first BN software tool to (1) break down BN construction and reasoning into specific steps that, combined with minimal upfront training and embedded help, can guide relatively novice users through the process; (2) support groups to collaboratively build a consensus BN, partly by implementing the entire process in an online platform; and (3) use the BN to produce a consensus written analytic report, assisted by a reasoning template and automatically generated key points. Initial experimental results, summarized in Section 4, are promising for both the usability and effectiveness of the BARD tool for assisting problem solving and reasoning, by both individuals and groups. The written analytic reports produced with BARD—29 by individuals, 145 by groups—were significantly better, assessed against problem-specific marking rubrics, than the controls.

While the version of the BARD tool presented here supports elicitation of all key elements of a BN, it lacks additional features that are available in other BN software packages, such as modeling with continuous variables, learning the structure from data, allowing the CPTs to be specified by equations, supporting decision-making more explicitly with decision and utility nodes,[29] and sensitivity analysis. We plan to enhance BARD with these features incrementally, utilizing the functionality of the existing BN software (Netica and AgenaRisk) used in BARD's back-end. Users can already import and export BNs in these back-end file formats; we plan to extend both back-end and conversion compatibility to other BN packages. Beyond industry-standard BN features, we plan to provide more support for BN idioms, such as those already designed for legal arguments (Lagnado et al., 2013). We also intend to incorporate some more technical methods into BARD to assist the facilitator in combining analysts' input: the statistical amalgamation of BNs (e.g., Flores et al., 2011), and group elicitation of vague parameters (e.g., Hanea et al., 2017).

The BARD platform provides a rich tool for research on how individuals and groups build and reason with BNs. BARD's configurable constraint-based structure layout will allow us to investigate whether particular enforced structure layouts improve understanding of a model and its reasoning, and/or aid comparisons between alternative BNs. We also intend to improve and test the efficacy of the various elements produced by our AET, and improve their presentation by making individual elements available on demand, and combining verbal with visual aids.[30]

BARD trades off the rigor of traditional Delphi for the flexibility and user-friendliness of a real-time version (both defined in Section 2.6). Piloting suggested that for our participants and tasks the trade-off is worthwhile. We also have some experimental evidence that even a minimal Delphi-style interaction improves the network structures produced by BARD groups (Bolger et al., 2020). Nevertheless, we do not yet have any direct experimental compari-

[28] Attrition was roughly 10% per week for the control and experimental participants alike, which is comparatively low for a Delphi study, but it inevitably accumulated over the six weeks, so that only 93 experimental participants (47%) completed this survey.

[29] A beta-version of BARD is now available that supports decision and utility nodes, and computes expected utilities for decisions.

[30] This quest for automated explanation of BNs has become a major three-year spinoff project, "Improving human reasoning with causal Bayes networks: a multimodal approach," involving several BARD researchers at Monash University and the University of London, and funded by the Australian Research Council. See https://dataportal.arc.gov.au/NCGP/Web/Grant/Grant/DP200100040.

son between BARD groups using traditional Delphi, real-time Delphi, and free interaction. Further research comparing social protocols is needed to measure their impact and optimize overall system performance. This investigation will be facilitated by the configurability of BARD user access, with three different versions of Delphi already available. Further possible research on collaboration includes investigating the best group size and the factors that may influence it (e.g., Belton et al., 2021); how a group may be split across different tasks; and how outputs from multiple groups working in parallel might be best combined, e.g., by a meta-level BARD group.

BARD is also well suited to educational applications, either to teach BN modeling or to use it as an analytic method for specific questions. Apart from the training and help it offers novices, it supports group work and assessment by documenting both individual contributions and group output, while allowing supervision by a moderator and observers. We are currently conducting an experiment comparing using BARD to using standard BN software for teaching BN modeling to undergraduate IT students.

DECLARATION OF COMPETING INTEREST

None.

ACKNOWLEDGMENTS

Funding for the BARD project was provided by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through their CREATE program under Contract 2017-16122000003.[31]

REFERENCES

Bayraktar, M. E., & Hastak, M. (2009). Bayesian belief network model for decision making in highway maintenance: Case studies. *Journal of Construction Engineering and Management*, *135*, 1357–1369. http://doi.org/10.1061/(ASCE)CO.1943-7862.0000111.

Belton, I., Wright, G., Sissons, A., Bolger, F., Crawford, M., Hamlin, I., … Vasilichi, A. (2021). The effect of Delphi group size and opinion diversity on participant experience. *OSF Preprints*, May 4 http://doi.org/10.31219/osf.io/e9ubm.

Blalock Jr., H. M. (1964). *Causal inferences in nonexperimental research*. Chapel Hill: The University of North Carolina Press.

[31]The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, *5*, 61–72.

Boerlage, B. (1992). Link strength in Bayesian networks. Master's thesis, University of British Columbia, Vancouver, Canada.

Bolger, F., Nyberg, E. P., Belton, I., Crawford, M., Hamlin, I., Taylor-Brown Lūka, C., … Wright, G. (2020). Improving the production and evaluation of structural models using a Delphi process. *OSF Preprints*, http://doi.org/10.31219/osf.io/v6qsp

Boneh, T. (2010). Ontology and Bayesian decision networks for supporting the meteorological forecasting process. Ph.D. thesis. Monash University, Melbourne, Australia.

Boneh, T., Weymouth, G., Newham, P., Potts, R., Bally, J., Nicholson, A., & Korb, K. (2015). Fog forecasting for Melbourne Airport using a Bayesian decision network. *Weather And Forecasting*, *30*, 1218–1233.

Bramley, N. R., Dayan, P., Griffiths, T. L., & Lagnado, D. A. (2017). Formalizing Neurath's ship: Approximate algorithms for online causal learning. *Psychological Review*, *124*(2017), 301–338. http://doi.org/10.1037/rev0000061

Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), *Usability evaluation in industry* (pp. 189–194). London: CRC Press.

Cawson, J., Hemming, V., Ackland, A., Anderson, W., Bowman, D., Bradstock, R.,… Penman, T. (2020). Exploring the key drivers of forest flammability in wet eucalypt forests using expert-derived conceptual models. *Landscape Ecology*, *35*, 1775–1798. http://doi.org/10.1007/s10980-020-01055-z

Charness, G., & Sutter, M. (2012). Groups make better self-interested decisions. *Journal of Economic Perspectives*, *26*, 157–176.

Chee, Y. E., Wilkinson, L., Nicholson, A. E., Quintana-Ascencio, P. F., Fauth, J. E., Hall, D., … Rumpff, L. (2016). Modelling spatial and temporal changes with GIS and spatial and dynamic Bayesian networks. *Environmental Modelling & Software*, *82*, 108–120. http://doi.org/10.1016/j.envsoft.2016.04.012

Choi, K.-H., Joo, S., Cho, S. I., & Park, J.-H. (2007). Locating intersections for autonomous vehicles: A Bayesian network approach. *ETRI Journal*, *29*, 249–251.

Chris, E. (1987). Explanation of probabilistic inference for decision support systems. In *Proceedings of the AAAI-87 Workshop on Uncertainty in Artificial Intelligence*, 394–403. Seattle, Washington.

Clapper, J. (2015). *Intelligence Community Directive (ICD) 203,Analytic Standards*. United States Office of the Director of National Intelligence (ODNI). https://www.dni.gov/files/documents/ICD/ICD%20203%20Analytic%20Standards.pdf.

Colson, A. R., & Cooke, R. M. (2018). Expert elicitation: Using the classical model to validate experts' judgments. *Review of Environmental Economics and Policy*, *12*, 113–132. http://doi.org/10.1093/reep/rex022

Cox, B. J., & Novobilski, A. (1991). *Object-oriented programming; An evolutionary approach* (2nd ed.). USA: Addison-Wesley.

Cruz, N., Desai, S. C., Dewitt, S., Hahn, U., Lagnado, D., Liefgreen, A., … Tešić, M. (2020). Widening access to Bayesian problem solving. *Frontiers in Psychology*, *11*, p. 660. http://doi.org/10.3389/fpsyg.2020.00660

Dwyer, T., Marriott, K., & Wybrow, M. (2009). Topology preserving constrained graph layout. In I. G. Tollis & M. Patrignani (Eds.), *Graph drawing* (pp. 230–241). Berlin: Springer.

Etminani, K., Naghibzadeh, M., & Peña, J. M. (2013). DemocraticOP: A democratic way of aggregating Bayesian network parameters. *International Journal of Approximate Reasoning*, *54*, 602–614.

Fenton, N., & Neil, M. (2000). The "Jury Fallacy" and the use of Bayesian networks to present probabilistic legal arguments. *Mathematics Today*, *37*, 61–102.

Fenton, N., Neil, M., & Lagnado, D. A. (2013). A general structure for legal arguments about evidence using Bayesian

networks. *Cognitive Science*, *37*, 61–102. http://doi.org/10.1111/cogs.12004

Flores, M., Nicholson, A., Brunskill, A., Korb, K., & Mascaro, S. (2011). Incorporating expert knowledge when learning Bayesian network structure: A medical case study. *Artificial Intelligence in Medicine*, *53*, 181–204.

van der Gaag, L. C., Renooij, S., Witteman, C. L. M., Aleman, B. M. P., & Taal, B. G. (1999). How to elicit many probabilities. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence,* UAI'99 pp. 647–654. San Francisco: Morgan Kaufmann.

Gigerenzer, G., & Hoffrage, U. (1995). How to improve Bayesian reasoning without instruction: Frequency formats. *Psychological Review*, *102*, 684.

Gopnik, A., Sobel, D. M., Schulz, L. E., & Glymour, C. (2001). Causal learning mechanisms in very young children: Two-, three-, and four-year-olds infer causal relations from patterns of variation and covariation. *Developmental Psychology*, *37*, 620.

Gordon, T., & Pease, A. (2006). RT Delphi: An efficient, "roundless" almost real time Delphi method. *Technological Forecasting and Social Change*, *73*, 321–333.

Hahn, U. (2014). The Bayesian boom: Good thing or bad? *Frontiers in Psychology*, *5*, 765.

Hahn, U., & Harris, A. J. (2014). What does it mean to be biased: Motivated reasoning and rationality. In B. H. Ross (Ed.), *Psychology of learning and motivation* (Vol. *61*, pp. 41–102). Cambridge, MA: Academic Press. http://doi.org/10.1016/B978-0-12-800283-4.00002-2

Hahn, U., & Oaksford, M. A. (2006). Bayesian approach to informal argument fallacies. *Synthese*, *152*, 207–236.

Hanea, A., McBride, M., Burgman, M., Wintle, B., Fidler, F., Flander, L., … Mascaro, S. (2017). Investigate discuss estimate aggregate for structured expert judgement. *International Journal of Forecasting*, *33*, 267–279. http://doi.org/10.1016/j.ijforecast.2016.02.008

Hemming, V., Burgman, M., Hanea, A., McBride, M., & Wintle, B. (2018). A practical guide to structured expert elicitation using the IDEA protocol. *Methods in Ecology and Evolution*, *9*, 169–180. http://doi.org/10.1111/2041-210X.12857

Jarvstad, A., & Hahn, U. (2011). Source reliability and the conjunction fallacy. *Cognitive Science*, *35*, 682–711.

Jitnah, N., Zukerman, I., McConachy, R., & George, S. (2000). Towards the generation of rebuttals in a Bayesian argumentation system. In *INLG'2000: Proceedings of the First International Conference on Natural Language Generation*, 39–46. Mitzpe Ramon, Israel: Association for Computational Linguistics. http://doi.org/10.3115/1118253.1118260.

Kahneman, D., Slovic, P., & Tversky, A. (Eds.) (1982). *Judgment under uncertainty: Heuristics and biases*. New York: Cambridge University Press.

Keppens, J. (2011). On extracting arguments from Bayesian network representations of evidential reasoning. In *Proceedings of the 13th International Conference on Artificial Intelligence and Law* ICAIL '11, pp. 141–150. New York: ACM.

Korb, K., McConachy, R., & Zukerman, I. (1997). A cognitive model of argumentation. In *Proceedings of the 19th Annual Conference of the Cognitive Science Society*, 400–405.

Korb, K. B. (2004). Bayesian informal logic and fallacy. *Informal Logic*, *24*, 41–70.

Korb, K. B., Geard, N., & Dorin, A. (2013). A Bayesian approach to the validation of agent-based models. In A. Tolk (Ed.), *Ontology, epistemology, and teleology for modeling and simulation* (pp. 255–269). Berlin: Springer-Verlag. http://doi.org/10.1007/978-3-642-31140-6_14.

Korb, K. B., & Nicholson, A. E. (2011). *Bayesian artificial intelligence*. (2nd ed). Boca Raton, FL: Chapman & Hall/CRC Press. [Volume of *Computer Science & Data Analysis*].

Korb, K. B., & Nyberg, E. P. (2016). Analysing arguments using causal Bayesian networks. *Bayesian Watch*, March 30. Retrieved from: https://bayesianwatch.wordpress.com/2016/03/30/aaucbn

Korb, K. B., Nyberg, E. P., & Hope, L. (2011). A new causal power theory. In P. M. Illari, F. Russo, & J. Williamson (Eds.), *Causality in the sciences* (pp. 628–652). Oxford: Oxford University Press.

Korb, K. B., Nyberg, E. P., Oshni Alvandi, A., Thakur, S., Ozmen, M., Li, Y., … Nicholson, A. E. (2020). Individuals vs. BARD: Experimental evaluation of an online system for structured, collaborative Bayesian reasoning. *Frontiers in Psychology*, *11*, 1054. http://doi.org/10.3389/fpsyg.2020.01054

Kugler, T., Kausel, E. E., & Kocher, M. G. (2012). Are groups more rational than individuals? A review of interactive decision making in groups. *Wiley Interdisciplinary Reviews: Cognitive Science*, *3*, 471–482.

Kushnir, T., Gopnik, A., Lucas, C., & Schulz, L. (2010). Inferring hidden causal structure. *Cognitive Science*, *34*, 148–160.

Kyrimi, E., & Marsh, W. (2016). A progressive explanation of inference in "hybrid" Bayesian networks for supporting clinical decision making. In A. Antonucci, G. Corani, & C. P. de Campos (Eds.), *Probabilistic Graphical Models - Eighth International Conference (PGM 2016), Lugano, Switzerland, September 6–9, 2016. Proceedings*, pp. 275–286. [JMLR.org, *52* of JMLR Workshop and Conference Proceedings.]

Lagnado, D. A., Fenton, N., & Neil, M. (2013). Legal idioms: A framework for evidential reasoning. *Argument & Computation*, *4*, 46–63.

Lagnado, D. A., & Gerstenberg, T. (2017). Causation in legal and moral reasoning. In M. R. Waldmann (Ed.), *Oxford handbook of causal reasoning* (pp. 565–602). Oxford: Oxford University Press. http://doi.org/10.1093/oxfordhb/9780199399550.013.30

Lagnado, D. A., & Sloman, S. (2004). The advantage of timely intervention. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*, 856.

Lagnado, D. A., & Sloman, S. A. (2006). Time as a guide to cause. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *32*, 451.

Langseth, H., & Portinale, L. (2007). Bayesian networks in reliability. *Reliability Engineering and System Safety*, *92*, 92–108.

Laskey, K. B., & Mahoney, S. M. Network fragments: Representing knowledge for constructing probabilistic models. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence,* UAI'97 pp. 334–341. San Francisco: Morgan Kaufmann.

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

Lichtenstein, S., Fischhoff, B., & Phillips, L. (1982). Calibrations of probabilities: The state of the art to 1980. In D. Kahneman, P. Slovic, & A. Tversky (Eds.), *Judgment under uncertainty: Heuristics and biases* (pp. 306–334). New York: Cambridge University Press.

Liefgreen, A., Tešić, M., & Lagnado, D. (2018). Explaining away: significance of priors, diagnostic reasoning, and structural complexity. In T. Roger, M. Rau, X. Zhu, & W. Kalish (Eds.), *Proceedings of the 40th Annual Conference of the Cognitive Science Society* (pp. 2044–2049). Austin, TX: Cognitive Science Society.

Linstone, H., & Turoff, M. (1975). *The Delphi method: Techniques and applications*. London: Addison-Wesley.

Malcolm, D. G., Roseboom, C. E., Clark, C. E., & Fazar, W. (1959). Application of a technique for research and development program evaluation. *Operations Research*, *7*, 646–649.

Mascaro, S., Nicholson, A. E., & Korb, K. B. (2014). Anomaly detection in vessel tracks using Bayesian networks. *International Journal of Approximate Reasoning*, *55*, 84–98. http://doi.org/10.1016/j.ijar.2013.03.012. [Special issue: *Applications of Bayesian Networks*.]

Misirli, A. T., & Bener, A. B. (2014). Bayesian networks for evidence-based decision-making in software engineering. *IEEE Transactions on Software Engineering*, *40*, 533–554.

Moore, D. A., & Healy, P. J. (2008). The trouble with overconfidence. *Psychological Review*, *115*, 502–517.

Mumford, M. D., Blair, C., Dailey, L., Leritz, L. E., & Osburn, H. K. (2006). Errors in creative thought? Cognitive biases in a complex processing activity. *The Journal of Creative Behavior*, *40*, 75–109.

Newell, B. R., Lagnado, D. A., & Shanks, D. R. (2015). *Straight choices: The Psychology of Decision Making*. (2nd ed.) Hove: Psychology Press.

Nicholson, A., Boneh, T., Wilkin, T., Stacey, K., Sonenberg, L., & Steinle, V. (2001). A case study in knowledge discovery and elicitation in an intelligent tutoring application. In P. Brusilovsky, A. Corbett, & F. de Rosis (Eds.), *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, UAI'01 pp. 386–394. Berlin: Springer.

Nicholson, A., Woodberry, O., Mascaro, S., Korb, K., Moorrees, A., & Lucas, A. (2011). ABC-BN: A tool for building, maintaining and using Bayesian networks in an environmental management application. In *Proceedings of the 8th Bayesian Modelling Applications Workshop* (Vol. *818*, pp. 331–335). http://ceur-ws.org/Vol-818.

Nicholson, A. E., Mascaro, S., Thakur, S., Korb, K. B., & Ashman, R. (2016). Delphi elicitation for strategic risk assessment. Technical Report TR-2016 Bayesian Intelligence Pty Ltd. https://bayesian-intelligence.com/publications/TR2016_1_Delphi_Elicitation.pdf.

Novak, J. (2010). *Learning, creating, and using knowledge: Concept maps as facilitative tools in schools and corporations*. (2nd ed.). New York: Routledge. http://doi.org/10.4324/9780203862001.

O'Donnell, R. T., Allison, L., & Korb, K. B. (2006). Learning hybrid Bayesian networks by MML. In A. Sattar, & B. Kang (Eds.), *AI 2006: Advances in artificial intelligence* (pp. 192–203). Berlin: Springer.

O'Hagan, A. (2019). Expert knowledge elicitation: Subjective but scientific. *The American Statistician*, *73*, 69–81. http://doi.org/10.1080/00031305.2018.1518265.

Packer, D. J. (2009). Avoiding groupthink: Whereas weakly identified members remain silent, strongly identified members dissent about collective problems. *Psychological Science*, *20*, 546–548.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo: Morgan Kaufmann.

Pearl, J., & Mackenzie, D. (2018). *The book of why: The new science of cause and effect* (1st ed). New York: Basic Books.

Pilditch, T., Hahn, U., & Lagnado, D. (2018). Integrating dependent evidence: Naïve reasoning in the face of complexity. In T. Roger, M. Rau, X. Zhu, & W. Kalish (Eds.), *Proceedings of the 40th Annual Conference of the Cognitive Science Society* (pp. 884–889). Austin, TX: Cognitive Science Society.

Pilditch, T. D., Fenton, N., & Lagnado, D. (2019). The zero-sum fallacy in evidence evaluation. *Psychological Science*, *30*, 250–260. http://doi.org/10.1177/0956797618818484

Pitchforth, J., & Mengersen, K. (2013). A proposed validation framework for expert elicited Bayesian networks. *Expert Systems with Applications*, *40*, 162–167. http://doi.org/10.1016/j.eswa.2012.07.026.

Rowe, G., Wright, G., & Bolger, F. (1991). Delphi: A reevaluation of research and theory. *Technological Forecasting and Social Change*, *39*, 235–251.

Salerno, J. M., Bottoms, B. L., & Peter-Hagene, L. C. (2017). Individual versus group decision making: Jurors' reliance on central and peripheral information to evaluate expert testimony. *PLOS One*, *12*, p. e0183580.

Sember, P., & Zukerman, I. (1989). Strategies for generating micro explanations for Bayesian belief networks. In *Proceedings of the Fifth Workshop on Uncertainty and Artificial Intelligence*, pp. 295–302. Windsor, Ontario.

Serwylo, P. (2015). Intelligently generating possible scenarios for emergency management during mass gatherings. Ph.D. thesis, Monash University, Melbourne, Australia.

Sesen, M. B., Nicholson, A. E., Banares-Alcantara, R., Kadir, T., & Brady, M. (2013). Bayesian networks for clinical decision support in lung cancer care. *PLOS One*, *8*, e82349. http://doi.org/10.1371/journal.pone.0082349

Sigurdsson, J. H., Walls, L. A., & Quigley, J. L. (2001). Bayesian belief nets for managing expert judgement and modelling reliability. *Quality and Reliability Engineering International*, *17*, 181–190. http://doi.org/10.1002/qre.410

Simon, H. (1954). Spurious correlation: A causal interpretation. *Journal of the American Statistical Association*, *49*, 467–479.

Sloman, S. A., & Lagnado, D. (2015). Causality in thought. *Annual Review of Psychology*, *66*, 223–247.

Speirs-Bridge, A., Fidler, F., McBride, M., Flander, L., Cumming, G., & Burgman, M. (2010). Reducing overconfidence in the interval judgments of experts. *Risk Analysis*, *30*, 512–523.

Spirtes, P., Glymour, C. N., & Scheines, R. (2000). *Causation, prediction, and search*. (2nd ed.) Cambridge, MA: MIT Press.

Stettinger, M., Felfernig, A., Leitner, G., & Reiterer, S. (2015). Counteracting anchoring effects in group decision making. In *23rd International Conference on User Modeling, Adaptation, and Personalization,* UMAP pp. 118–130. Cham: Springer.

Straus, S. G., Parker, A. M., & Bruce, J. B. (2011). The group matters: A review of processes and outcomes in intelligence analysis. *Group Dynamics: Theory, Research, and Practice*, *15*, 128.

Suermondt, H. J. (1992). Explanation in Bayesian belief networks. Ph.D. thesis, Stanford University, Palo Alto, California.

Tversky, A., & Kahneman, D. (1982). Evidential impact of base rates. In D. Kahneman, P. Slovic, & A. Tversky (Eds.), *Judgment under uncertainty: Heuristics and biases* (pp. 153–160). New York: Cambridge University Press.

Villejoubert, G., & Mandel, D. R. (2002). The inverse fallacy: An account of deviations from Bayes's theorem and the additivity principle. *Memory & Cognition*, *30*, 171–178. http://doi.org/10.3758/BF03195278

Vlek, C., Prakken, H., Renooij, S., & Verheij, B. (2016). A method for explaining Bayesian networks for legal evidence with scenarios. *Artificial Intelligence and Law*, *24*, 285–324. http://doi.org/10.1007/s10506-016-9183-4.

Vreeswijk, G. A. W. (2005). Argumentation in Bayesian belief networks. In I. Rahwan, P. Moraïtis, & C. Reed (Eds.), *Argumentation in multi-agent systems* (pp. 111–129). Berlin: Springer.

Welsh, M. B., & Navarro, D. J. (2012). Seeing is believing: Priors, trust, and base rate neglect. *Organizational Behavior and Human Decision Processes*, *119*, 1–14.

Wieten, R., Bex, F., Prakken, H., & Renooij, S. (2019). Constructing Bayesian network graphs from labeled arguments. In G. Kern-Isberner, & Z. Ognjanović (Eds.), *Symbolic and quantitative approaches to reasoning with uncertainty* (pp. 99–110). Cham: Springer. http://doi.org/10.1007/978-3-030-29765-7_9

Wintle, B. C., Fraser, H., Wills, B. C., Nicholson, A. E., & Fidler, F. (2019). Verbal probabilities: Very likely to be somewhat more confusing than numbers. *PLOS ONE*, *14*, (4), e0213522. https://doi.org/10.1371/journal.pone.0213522.

Wright, S. (1934). The method of path coefficients. *The Annals of Mathematical Statistics*, *5*, 161–215.

Zukerman, I., Herrmann, M., Azad, A., Nyberg, E. P., Mascaro, S., & Nicholson, A. E. (2019). Automated explanation of Bayesian network reasoning to support structured analysis. Technical Report TR-2019-1 Bayesian Intelligence Pty Ltd. https://bayesian-intelligence.com/publications/TR2019_1_Automated_Explanation.pdf

Zukerman, I., McConachy, R., & Korb, K. B. (1998). Bayesian reasoning in an abductive mechanism for argument generation and analysis. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence,* AAAI-98. Madison, Wisconsin. pp. 833–838.

Zukerman, I., McConachy, R., Korb, K. B., & Pickett, D. (1999). Exploratory interaction with a Bayesian argumentation system. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence,* IJCAI-99, Stockholm, Sweden. pp. 1294–1299.