This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

Exploiting Experts' Knowledge for Structure Learning of Bayesian Networks

Hossein Amirkhani, Mohammad Rahmati, Peter J.F. Lucas, and Arjen Hommersom

Abstract—Learning Bayesian network structures from data is known to be hard, mainly because the number of candidate graphs is super-exponential in the number of variables. Furthermore, using observational data alone, the true causal graph is not discernible from other graphs that model the same set of conditional independencies. In this paper, it is investigated whether Bayesian network structure learning can be improved by exploiting the opinions of multiple domain experts regarding cause-effect relationships. In practice, experts have different individual probabilities of correctly labeling the inclusion or exclusion of edges in the structure. The accuracy of each expert is modeled by three parameters. Two new scoring functions are introduced that score each candidate graph based on the data and experts' opinions, taking into account their accuracy parameters. In the first scoring function, the experts' accuracies are estimated using an expectation-maximization-based algorithm and the estimated accuracies are explicitly used in the scoring process. The second function marginalizes out the accuracy parameters to obtain more robust scores when it is not possible to obtain a good estimate of experts' accuracies. The experimental results on simulated and real world datasets show that exploiting experts' knowledge can improve the structure learning if we take the experts' accuracies into account.

Index Terms—Bayesian networks, structure learning, experts' knowledge, experts' accuracy, marginalization-based score

1 Introduction

Bayesian networks are a popular class of probabilistic graphical models that are applicable to many problems that are characterized by uncertainty concerning multiple variables and their relationships. At a qualitative level, the structure of a Bayesian network describes the relationships between random variables in the form of conditional independence relations. At a quantitative level, (local) relationships between random variables are described by (conditional) probability distributions, also called Bayesian network parameters. To apply Bayesian networks to a particular domain, it is first necessary to learn the Bayesian network structure and its parameters for that particular problem domain. Alternatively, one may design a Bayesian network structure based on experts' knowledge alone and then use either subjective estimates or statistical parameter estimation to obtain the Bayesian network. This paper focuses on the issue of how structure learning can benefit from available experts' knowledge.

During the last two decades, many Bayesian network structure learning algorithms have been proposed (e.g. [1], [2], [3], [4], [5], [6], [7]). One of the most widely used class of structure learning algorithms are the *score-based* methods. They attempt to identify the model that best fits the data by searching through the space of candidate models and selecting the one that obtains the highest score. The search is guided by various heuristics, very often hill-climbing [2], but genetic algorithms [8] and particle swarm optimization [9] have also been used. Typical scoring functions are the Akaike information criteria (AIC) [10], the Bayesian information criteria (BIC) [11], and the Bayesian Dirichlet equivalence uniform (BDeu) [2] scores. The other common approaches, often referred to as the *constraint-based* methods [12], estimate from the data whether certain conditional independencies between the variables hold. Networks that are consistent with these independencies are selected.

There are several challenges a Bayesian network structure learning algorithm encounters when trying to discover a good model. First, the number of candidate graphs is super-exponential in the number of variables. More precisely, the number of DAGs with n variables is greater than $2^{\binom{n}{2}}$ (i.e., the number of undirected graphs with n variables); the exact number of DAGs can be computed using Robinson's formula [13]. Because of the huge search space, the learning problem is hard [14]. This implies that for more than six variables, heuristic search is needed, and thus the globally optimal Bayesian network may not be found. This is complicated by the fact that in many practical learning settings, there is little data or the data are noisy, so that the score that is being used is not accurate. Furthermore, for most structures there are many different *Markov equivalent* graphs that encode the same independence relations, i.e., these structures cannot be distinguished based on data alone. These limitations generally lead to learned models that substantially differ from the true causal network of the underlying problem.

Given these limitations of Bayesian network structure

- *H. Amirkhani was with the Computer Engineering and Information Technology Department, Amirkabir University of Technology, Tehran, Iran. Presently, he is with the Technology and Engineering Department, University of Qom, Qom, Iran. E-mail:amirkhani@aut.ac.ir.*
- *M. Rahmati is with the Computer Engineering and Information Technology Department, Amirkabir University of Technology, Tehran, Iran. E-mail:rahmati@aut.ac.ir.*
- *P.J.F. Lucas is with the Institute for Computing and Information Sciences, Radboud University, Nijmegen, The Netherlands. E-mail: plucas@liacs.nl.*
- *A. Hommersom is with the Institute for Computing and Information Sciences, Radboud University, Nijmegen, The Netherlands. He is also with the Faculty of Science, Management & Technology, Open University, Heerlen, The Netherlands. E-mail: arjenh@cs.ru.nl.*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

learning, some researchers have proposed the use of experts' knowledge to bias the search procedure and reduce the complexity of the search space [1], [15], [16], [17], [18], [19]. A shortcoming in the majority of such methods is that they assume that there exists a completely reliable expert, and the expert's opinions about the structure are considered to be consistent with the true structure. It is obvious that in a real world setting, each expert may produce some errors in the provided opinions. In fact, it is more realistic to assume that we have to deal with multiple experts with varying levels of expertise rather than an omniscient expert.

In this paper, we propose two novel scoring functions to combine the available data with the knowledge from multiple, possibly unreliable, experts. The main advantages are that (i) it is not necessary to have a completely reliable expert, (ii) experts only have to label some of the edges (included in the graph, or not), and (iii) these scores can deal with conflicts between experts. In the first approach, we propose an expectation-maximization-based method for estimating the accuracy of each expert, then this information is explicitly used to score each structure based on both data and experts' opinions. In the second approach, we propose a Bayesian alternative by taking into account the uncertainty in the accuracy of each expert.

The first scoring function which is proposed in this paper, which we refer to as the *explicit-accuracy-based score*, builds upon the method originally proposed by [16]. The main advantage of our approach is that we assume that experts are heterogeneous, i.e., different experts have different levels of accuracy. In addition, with our second score, referred to as the *marginalization-based score*, we are able to handle the problem that the estimated experts' accuracies may not be so reliable, and we obtain a more robust score by marginalizing out the experts' accuracy parameters. Experimental results reveal that exploiting experts' knowledge can improve the structure learning if we take the experts' accuracies into account. Specifically, if the experts' accuracies can be confidently estimated, it is suggested to explicitly use the estimated accuracies in the scoring process, otherwise, marginalizing out the accuracy parameters yields more robust scores.

The rest of this paper is organized as follows. In Section 2, we introduce the notations and preliminaries that will be used in subsequent sections. Specially, we present a three-parameter-based model of experts' accuracies in this section. Then, in Section 3, we clarify our problem setting based on some graphical models. Sections 4 and 5 present our scoring functions, i.e. explicit-accuracy-based score and marginalization-based score, respectively. Section 6 details our experimental procedures and presents the results. Finally, Section 7 concludes the paper.

2 Preliminaries

In this section, we first introduce Bayesian networks and some Markov independence properties. Subsequently, we briefly review score-based Bayesian network structure learning. Finally, we present our three-parameter-based model of experts' accuracies, along with some further notations that will be used throughout the remainder of this paper.

2.1 Bayesian Networks

Formally, a *Bayesian network*, or BN for short, is a tuple $\mathcal{B} = (G, \mathcal{X}, P)$, with $G = (V, E)$ a directed acyclic graph (DAG) with set of nodes V and directed edges or arcs $E \subseteq V \times V$, $\mathcal{X} = \{X_1, \ldots, X_n\}$ is a set of random variables with a 1-1 correspondence to V, and P is a joint probability distribution over $\mathcal{X}$. An arc is denoted by $(X_i \rightarrow X_j) \in E$ or $(X_j \leftarrow X_i) \in E$. In the following we assume that the random variables X_i are all discrete. According to the *chain rule for Bayesian networks*, P can be written as the product of the probabilities of the random variables, conditioned on their parents:

$$P(X_1, \ldots, X_n) \quad = \quad \prod_{i=1}^{n} P(X_i \mid \pi(X_i)),$$

where $\pi(X_i)$ is the set of parents of X_i, i.e., the set $\{X_j \mid (X_j \rightarrow X_i) \in E\}$. The number of values that these parents can take is denoted by $q_i = \prod_{X_j \in \pi(X_i)} r_j$, where r_j is the number of values that X_j can take.

The graph structure of G encodes a set of independence assumptions about P which is formalized by *d-separation* (directed separation): If a set of nodes X d-separates another set of nodes Y given a set of nodes Z, then X is independent of Y given Z, written as $(X \perp Y \mid Z)$. D-separation is defined as follows. Let ρ be a trail in G, i.e., a path without considering the directions of the arcs. A trail ρ is said to be *blocked* by a set of nodes Z if and only if (at least) one of the following holds:

- ρ contains a *chain* $U \rightarrow Z_i \rightarrow W$, such that Z_i is in Z,
- ρ contains a *fork* $U \leftarrow Z_i \rightarrow W$, such that Z_i is in Z,
- ρ contains a *collider* $U \rightarrow Z_i \leftarrow W$, such that neither Z_i nor any descendant of Z_i is in Z.

Then, X and Y are said to be *d-separated* by Z if any trail between any node in X and any node in Y is blocked by Z. One of the special features of a Bayesian network is that, through the notion of collider, variables that are (conditionally) independent could become dependent by conditioning on the collider or one of its descendants. An undirected network, i.e., Markov random field, does not have this property [20].

If two graphs encode the same set of independencies, then we say that these graphs are *Markov equivalent*. To represent equivalence classes of DAGs, partially directed acyclic graphs (PDAGs) are employed, which are acyclic graphs with both directed and undirected edges. The *completed PDAG* (CPDAG) [21] – also called the *essential graph* [22] – of a DAG $G = (V, E)$ is a PDAG G' such that (i) it contains the same nodes as G, and (ii) for each edge $(X \rightarrow Y) \in E$, if each graph in the equivalence class of G has the edge $X \rightarrow Y$, then $X \rightarrow Y$ is in G'; otherwise $X - Y$ is in G'. The consequence of the definition of Markov equivalence is that some arcs have a strict orientation and meaning, whereas in others the orientation can also be reversed without changing the meaning.

2.2 Score-Based Structure Learning

The structure learning task is basically to find the graph, or the structure, that fits the data best. In this paper, we employ a score-based approach that attempts to identify

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

the appropriate model by a hill-climbing search through the space of candidate models and selecting the one with the highest score [2]. The hill-climbing search selects at each step the best transformation among all feasible edge removals, edge reversals, and edge additions. Obviously, it ignores the edge reversals and edge additions that create directed cycles in the graph. When the score cannot be strictly improved anymore, the search stops.

Bayesian network scores are usually based on a maximum likelihood principle that picks the model that best 'fits' the observed data. To prevent overfitting, a Bayesian Occam's razor [23] can be used to select the model with the highest *marginal* likelihood $P(D \mid G)$, i.e., where the parameters are integrated out:

$$P(D \mid G) = \int P(D \mid G, \Theta) f(\Theta \mid G) d\Theta,$$

with f a probability density, such that Θ are the possible parameters for DAG G. Assuming that the conditional distributions defined in a Bayesian network are independent, [2] showed that this implies that the prior of these conditional distributions must be a Dirichlet, i.e., $\theta_{ij} \sim \mathrm{Dir}(\alpha_{ij})$, where θ_{ij} represents $P(X_i \mid \pi(X_i) = j)$ such that j is one of the configuration of the parents of X_i, $1 \leq j \leq q_i$, and α_{ij} is a vector of length r_i. Let N_{ijk} be the counts of $X_i = k$ and its parents having the value j in the data D, and $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$, it can then be shown that:

$$P(D \mid G) = \prod_{i=1}^{|V|} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})},$$

where $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ is the Gamma function. This score is called the BD (Bayesian Dirichlet) score. [2] also proved that for complete graphs, the only prior that assigns the same marginal likelihood to Markov equivalent graphs is the prior where:

$$\alpha_{ijk} = \alpha P_0(X_i = k, \pi(X_i) = j)$$

with $\alpha > 0$, where P_0 is a prior distribution. Finally, taking a uniform prior for P_0, i.e., $P_0(X_i = k, \pi(X_i) = j) = \frac{1}{q_i r_i}$, we obtain a very popular score called the BDeu (Bayesian Dirichlet equivalent uniform) score. In this score, the only parameter which we need to choose is α, which is also referred to as the *equivalent sample size*.

2.3 Edge Types, Experts' Opinions and Accuracies

If the number of nodes in the structure is n, i.e., $n = |V|$, then there are $N = n(n-1)/2$ different node pairs. Throughout this paper we assume that there is a fixed ordering over node pairs, and a fixed ordering over the nodes in each pair. If the ith pair is (X, Y), the status of the edge between X and Y is indicated by g_i, where

- $g_i = \rightarrow$ if $(X \rightarrow Y) \in E$,
- $g_i = \leftarrow$ if $(X \leftarrow Y) \in E$,
- $g_i = \nleftrightarrow$ if neither $(X \rightarrow Y)$ nor $(X \leftarrow Y)$ is in E.

According to the above notations, there are three edge types in the structure: $\{\rightarrow, \leftarrow, \nleftrightarrow\}$. Note that the edge types $\rightarrow$ and $\leftarrow$ do not essentially differ, but depend on the ordering over the nodes in the pairs. As an example, consider the Bayesian network structure depicted in

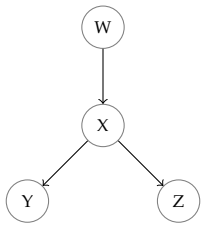


Fig. 1. Simple Bayesian network structure.

Fig. 1. Since there are 4 nodes in this graph, the number of node pairs is $N = 6$. If these pairs are ordered as (X,Y), (X,Z), (X,W), (Y,Z), (Y,W), (Z,W), we have $g_1 = \rightarrow$, $g_2 = \rightarrow$, $g_3 = \leftarrow$, $g_4 = \nleftrightarrow$, $g_5 = \nleftrightarrow$, $g_6 = \nleftrightarrow$.

We denote the prior distribution over edge types as $\boldsymbol{p} = \{p_\rightarrow, p_\leftarrow, p_\nleftrightarrow\}$. For example, when $\boldsymbol{p} = \{p_\rightarrow = 0.1, p_\leftarrow = 0.2, p_\nleftrightarrow = 0.7\}$ it means that prior to having any data or experts' knowledge, we believe that 10%, 20%, and 70% of g_is are respectively equal to $\rightarrow$, $\leftarrow$, $\nleftrightarrow$.

The number of experts is indicated by R. The opinion of the ith expert regarding the jth pair is denoted by $O_j^i \in \{\emptyset, \rightarrow, \leftarrow, \nleftrightarrow\}$, where $O_j^i = \emptyset$ meaning that the ith expert has not provided any opinion about the jth pair. We use O^i to indicate all opinions provided by the ith expert, O_j to mention the opinions provided by all experts about the jth pair, and O to denote all provided opinions.

We model the *accuracy of an expert* by three parameters:

- γ_1: The probability of detecting the existing edges with *correct* directions,
- γ_2: The probability of detecting the existing edges with *reverse* directions,
- γ_3: The probability of correctly detecting the *absent* edges.

We add a superscript such as $\gamma_1^i, \gamma_2^i, \gamma_3^i$ to denote the accuracy parameters of the ith expert. In addition, γ^i indicates the set containing all three accuracy parameters of the ith expert. Finally, the accuracy parameters of all experts are collectively denoted by boldface $\boldsymbol{\gamma}$.

As an example assume that the accuracy parameters of the ith expert are $\gamma_1^i = 0.6$, $\gamma_2^i = 0.1$, $\gamma_3^i = 0.8$. We can conclude that if this expert gives an opinion about the jth pair, the following confusion matrix shows the probabilities of providing different opinions by this expert:

$$\begin{array}{c} \\ \rightarrow \\ \leftarrow \\ \nleftrightarrow \end{array} \begin{array}{c} \begin{array}{ccc} \rightarrow & \leftarrow & \nleftrightarrow \end{array} \\ \begin{pmatrix} 0.6 & 0.1 & 0.3 \\ 0.1 & 0.6 & 0.3 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \end{array},$$

where each row shows a possible value for g_j and each column indicates a possible opinion O_j^i. Obviously, each row must sum to one. About the last row note that when the expert is wrong about the absent edge $g_j = \nleftrightarrow$, he/she selects one of the possible edges $\rightarrow$ or $\leftarrow$. We consider these two possibilities equally likely because we do not have any evidence to favor one over the other.

3 Problem Setting

Figs. 2 to 4 depict the various problem component models. Fig. 2 shows the factors affecting the structure G, data D,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

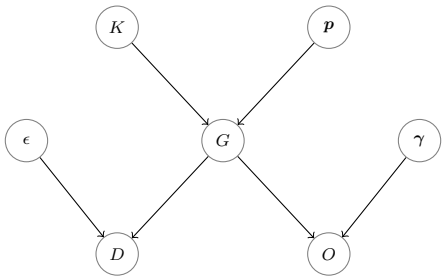


Fig. 2. Graphical model of the factors affecting the structure, data, and experts' opinions (G: graph structure; D: data; O: experts' opinions; $\boldsymbol{p}$: prior distribution over edge types; K: information that determines G; $\boldsymbol{\gamma}$: accuracy of experts; ϵ: noise). See text for further explanation.

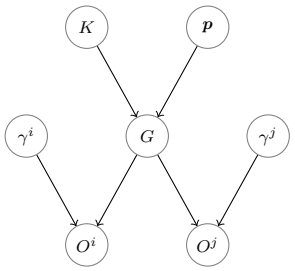
and experts' opinions O. In this model, K denotes the set of information determining the structure. Note that the prior distribution $\boldsymbol{p}$ can be seen as a part of K, but we separate it because it plays a distinguished role in the next section. According to this model, data is directly affected by the structure and a noise factor ϵ. Experts' opinions are also directly affected by the graph structure and experts' accuracy parameters.

Fig. 3 shows graphical models of the experts' opinions. The right model represents a more detailed version of the left one. According to this figure, the opinions of each expert are determined by the graph structure and the individual's accuracy parameters. More precisely, according to the detailed model, the opinion of one expert regarding one particular node pair is influenced by the edge status of that pair and the experts' accuracy parameters.

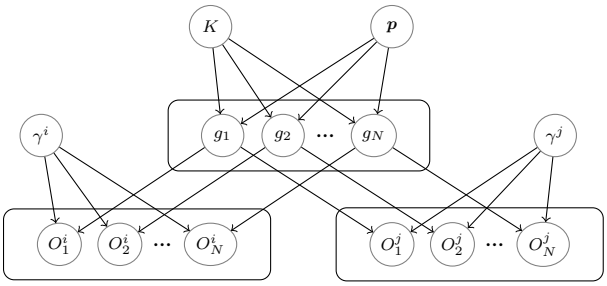
Fig. 4 indicates the roles of different accuracy parameters of an expert in determining the personal opinions. In this figure, O^i_E and O^i_A are the opinions of the ith expert about the existing and absent edges in G, respectively. Based on this model, the opinions regarding existing edges are influenced by γ_1 and γ_2 parameters, and the opinions regarding absent edges are influenced by the γ_3 parameter.

Using the d-separation rules in Bayesian networks, introduced in Section 2, we can derive a set of conditional independence statements from these models. Some of these statements are presented in Table 1. Only those used in the remainder of the paper are listed here, as clearly, more statements can be read off from the graphical models. In each of the statements in Table 1, assume that $1 \leq i, j \leq R$, $1 \leq x, y \leq N$, $i \neq j$, and $x \neq y$.

There is one point that must be noted about the independence statements in Table 1. Consider statement 3 as an example. According to this statement, D and $\boldsymbol{\gamma}$ are independent given G. Note that based on Fig. 2, if O is not given, D and $\boldsymbol{\gamma}$ are independent, regardless of whether G is given or not. Anyway, having G does not violate this independence, and because we need $D \perp \boldsymbol{\gamma} \mid G$ in the subsequent sections, we introduce this statement instead of $D \perp \boldsymbol{\gamma}$. This also holds for some other statements in Table 1.



(a) Abstract model



(b) Detailed model

Fig. 3. Graphical models of the experts' opinions.

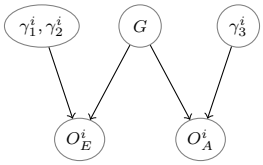


Fig. 4. Graphical model of the roles of different accuracy parameters.

TABLE 1
Some Independence Statements Derived from Models of Figs. 2 to 4

Number	Statement	Model
1	$G \perp \boldsymbol{\gamma}$	Fig. 2
2	$G \perp \boldsymbol{\gamma} \mid \boldsymbol{p}$	Fig. 2
3	$D \perp \boldsymbol{\gamma} \mid G$	Fig. 2
4	$O \perp D \mid G$	Fig. 2
5	$O \perp D \mid G, \boldsymbol{\gamma}$	Fig. 2
6	$O^i \perp O^j \mid G$	Fig. 3a
7	$O^i \perp O^j \mid G, \boldsymbol{\gamma}$	Fig. 3a
8	$O^j \perp \gamma^i \mid G, \gamma^j$	Fig. 3a
9	$O^j_x \perp O^j_y \mid G, \gamma^j$	Fig. 3b
10	$O^j_x \perp g_y \mid g_x, \gamma^j$	Fig. 3b
11	$O^i_x \perp O^j_x \mid g_x, \boldsymbol{p}, \boldsymbol{\gamma}$	Fig. 3b
12	$O^j_x \perp \gamma^i \mid g_x, \boldsymbol{p}, \gamma^j$	Fig. 3b
13	$O^j_x \perp \boldsymbol{p} \mid g_x, \gamma^j$	Fig. 3b
14	$(\gamma^i_1, \gamma^i_2) \perp \gamma^i_3$	Fig. 4

4 Explicit-Accuracy-Based Score

In our first scoring function, we explicitly use the estimated accuracy parameters of experts to quantify the quality of candidate structures. In subsection 4.1, we introduce this scoring function assuming that there is an estimate of experts' accuracies. Then, in subsection 4.2, an expectation-maximization-based method is described to estimate these parameters.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

4.1 Score Derivation

The goal of the explicit-accuracy-based scoring function is to score a candidate structure G using the data D, experts' opinions O, and estimated experts' accuracies $\boldsymbol{\gamma}$. A Bayesian measure of the goodness of fit of G is its posterior probability given D, O, and $\boldsymbol{\gamma}$:

$$P(G \mid D, O, \boldsymbol{\gamma}) \propto P(G, D, O, \boldsymbol{\gamma}) = P(\boldsymbol{\gamma})\, P(G \mid \boldsymbol{\gamma})\, P(D \mid G, \boldsymbol{\gamma})\, P(O \mid G, D, \boldsymbol{\gamma}).$$

Since $P(\boldsymbol{\gamma})$ does not depend on the graph structure, we omit it from the score. $P(G \mid \boldsymbol{\gamma})$ is simplified to $P(G)$ based on statement 1 in Table 1. In addition, $P(D \mid G, \boldsymbol{\gamma})$ is simplified to $P(D \mid G)$ according to statement 3. Finally, $P(O \mid G, D, \boldsymbol{\gamma})$ is simplified to $P(O \mid G, \boldsymbol{\gamma})$ using statement 5. Therefore, the explicit-accuracy-based score is introduced as the log of $P(G \mid D, O, \boldsymbol{\gamma})$ and given as:

$$\text{Score}_{explicit}(G; D, O, \boldsymbol{\gamma}) = \log P(G) + \log P(D \mid G) + \log P(O \mid G, \boldsymbol{\gamma}). \quad (1)$$

For the first two parts of this score, there are different choices mentioned in the literature. For the prior $P(G)$, the simplest and most common choice, which we also use in our experiments, is the uniform prior. It means that all structures are equally likely a priori, and therefore we can omit it from the score. Other choices are to provide greater penalty to dense networks [24] and to consider the number of options in determining the parents of each node [25]. For the second part $\log P(D \mid G)$, we use the BDeu score introduced in Section 2.

For the last part of the explicit-accuracy-based score, we use statements 7, 8, 9, 10 from Table 1 and obtain

$$\log P(O \mid G, \boldsymbol{\gamma}) = \sum_{j=1}^{R} \sum_{i=1}^{N} \log P(O_i^j \mid g_i, \gamma^j). \quad (2)$$

The term $P(O_i^j \mid g_i, \gamma^j)$ in equation (2) is computed using the decision tree depicted in Fig. 5. Note that we only use the provided opinions (i.e., $O_i^j \neq \emptyset$) to score G. The reason for dividing $\left(1 - \gamma_3^j\right)$ by 2 in this figure is that when the expert is wrong about an absent edge in G like $X \nleftrightarrow Y$, he/she selects one of the possible edges $X \rightarrow Y$ or $X \leftarrow Y$. We consider these two possibilities equally likely because there is no evidence to favor one over the other.

4.2 Expectation-Maximization-Based Accuracy Estimation

One way to estimate the experts' accuracies is to use the expectation-maximization (EM) algorithm [26]. This approach has been used in the crowdsourcing literatures such as [27], [28]. In the structure learning problem, we also previously used this algorithm to estimate the experts' confusion matrices [29], [30]. Here, we follow the same framework but derive the formulas for the three-parameter-based model of experts' accuracies proposed in Section 2.

If we consider the prior distribution $\boldsymbol{p}$ and the experts' accuracies $\boldsymbol{\gamma}$ as the parameters, the maximum-likelihood estimate of these parameters is

$$(\hat{\boldsymbol{p}}, \hat{\boldsymbol{\gamma}}) = \arg\max_{\boldsymbol{p}, \boldsymbol{\gamma}} \{\log P(O \mid \boldsymbol{p}, \boldsymbol{\gamma})\}.$$

Note that we use only the experts' opinions O in the likelihood function. Obviously, the data D can also help in this estimation problem, but we ignore it for simplicity's sake.

To solve this optimization problem, we consider the true structure G as a hidden variable and use the EM algorithm. We assume that (O_i, g_i) is independent of (O_j, g_j), for each $j \neq i$, given $(\boldsymbol{p}, \boldsymbol{\gamma})$. This assumption is not true in general, but to make the computations tractable, we ought to consider it. With this assumption, the log-likelihood of complete data (O, G) is

$$\log P(O, G \mid \boldsymbol{p}, \boldsymbol{\gamma}) = \sum_{i=1}^{N} \log P(O_i, g_i \mid \boldsymbol{p}, \boldsymbol{\gamma}).$$

Since g_i is a member of $\{\rightarrow, \leftarrow, \nleftrightarrow\}$, we can write

$$\log P(O_i, g_i \mid \boldsymbol{p}, \boldsymbol{\gamma}) = \sum_{k \in \{\rightarrow, \leftarrow, \nleftrightarrow\}} I(g_i = k) \log P(O_i, g_i = k \mid \boldsymbol{p}, \boldsymbol{\gamma}),$$

where $I(c)$ is the indicator function, which is one if the condition c is satisfied and zero otherwise. We simply expand the inner term in the above expression as

$$P(O_i, g_i = k \mid \boldsymbol{p}, \boldsymbol{\gamma}) = P(g_i = k \mid \boldsymbol{p}, \boldsymbol{\gamma}) P(O_i \mid g_i = k, \boldsymbol{p}, \boldsymbol{\gamma}).$$

Using statement 2 in Table 1, we have

$$P(g_i = k \mid \boldsymbol{p}, \boldsymbol{\gamma}) = P(g_i = k \mid \boldsymbol{p}) = p_k. \quad (3)$$

Also according to statement 11 in Table 1, we have

$$P(O_i \mid g_i = k, \boldsymbol{p}, \boldsymbol{\gamma}) = \prod_{j=1}^{R} P(O_i^j \mid g_i = k, \boldsymbol{p}, \boldsymbol{\gamma}). \quad (4)$$

Finally, based on statements 12 and 13 in Table 1, the term in the above equation is simplified to

$$P(O_i^j \mid g_i = k, \boldsymbol{p}, \boldsymbol{\gamma}) = P(O_i^j \mid g_i = k, \gamma^j),$$

which is simply computed using the decision tree in Fig. 5. Again, note that we consider only the provided opinions (i.e., $O_i^j \neq \emptyset$) in the computations.

Putting all the above together, the log-likelihood of complete data is

$$\log P(O, G \mid \boldsymbol{p}, \boldsymbol{\gamma}) = \sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow, \nleftrightarrow\}} \Big\{ I(g_i = k) \times \Big(\log p_k + \sum_{j=1}^{R} \log P(O_i^j \mid g_i = k, \gamma^j) \Big) \Big\}. \quad (5)$$

The EM algorithm iterates between two steps: an Expectation step (E-step) and a Maximization step (M-step). In the E-step, the expectation of the complete log-likelihood is computed using the current estimate of parameters. In the M-step, this expectation is maximized to obtain the next estimate of the parameters.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

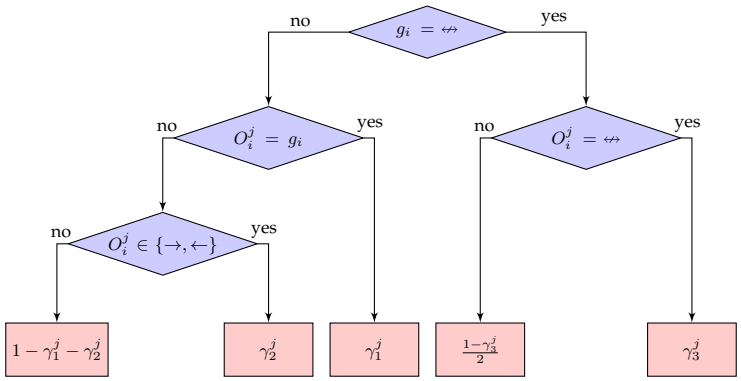


Fig. 5. Decision tree for computing $P(O_i^j \mid g_i, \gamma^j)$ for $O_i^j \neq \emptyset$.

The conditional expectation of the complete log-likelihood (5) given opinions O and the current estimate of the parameters $\boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)}$ is

$$\mathbb{E}_\eta[\log P(O, G \mid \boldsymbol{p}, \boldsymbol{\gamma})] = \sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow, \nleftrightarrow\}} \Bigg\{ \mathbb{E}_\eta[I(g_i = k)] \times \Bigg(\log p_k + \sum_{j=1}^{R} \log P(O_i^j \mid g_i = k, \gamma^j) \Bigg) \Bigg\}, \quad (6)$$

where we denote $\mathbb{E}_{G|O,\boldsymbol{p}^{(t)},\boldsymbol{\gamma}^{(t)}}$ by $\mathbb{E}_\eta$ for the ease of reading.

The expectation of the indicator function is the probability of the associated event. Therefore,

$$\mathbb{E}_\eta[I(g_i = k)] = P(g_i = k \mid O, \boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)}).$$

To make the computation of the above probability tractable, we assume that

$$P(g_i = k \mid O, \boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)}) = P(g_i = k \mid O_i, \boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)}).$$

This informally means that the status of the edge between two particular nodes is independent of the opinions regarding other node pairs, given the opinions about that node pair. This assumption means that that experts offer opinions about individual edges without taking opinions about other edges into account. It is based on assuming limited understanding of the semantics of Bayesian networks, quite common in domain experts of real-life problems.

Based on the above assumption and using Bayes' rule, we have

$$\mathbb{E}_\eta[I(g_i = k)] = \frac{P(g_i = k \mid \boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)}) \times P(O_i \mid g_i = k, \boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)})}{P(O_i \mid \boldsymbol{p}^{(t)}, \boldsymbol{\gamma}^{(t)})}. \quad (7)$$

The numerator terms can be computed using equations (3) and (4), respectively, and the denominator is simply a normalization factor.

The next estimates of parameters are obtained by maximizing the expectation (6). By setting the partial derivatives of (6) with respect to each parameter equal to zero, we obtain the following estimates for the parameters:

$$\begin{aligned}
p_k^{(t+1)} &= \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_\eta[I(g_i = k)], \\
(\gamma_1^j)^{(t+1)} &= \frac{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} \mathbb{E}_\eta[I(g_i = k)] \times I(O_i^j = k)}{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} \mathbb{E}_\eta[I(g_i = k)] \times I(O_i^j \neq \emptyset)}, \\
(\gamma_2^j)^{(t+1)} &= \frac{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} \mathbb{E}_\eta[I(g_i = k)] \times I(O_i^j \neq k, O_i^j \in \{\rightarrow, \leftarrow\})}{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} \mathbb{E}_\eta[I(g_i = k)] \times I(O_i^j \neq \emptyset)}, \\
(\gamma_3^j)^{(t+1)} &= \frac{\sum_{i=1}^{N} \mathbb{E}_\eta[I(g_i = \nleftrightarrow)] \times I(O_i^j = \nleftrightarrow)}{\sum_{i=1}^{N} \mathbb{E}_\eta[I(g_i = \nleftrightarrow)] \times I(O_i^j \neq \emptyset)},
\end{aligned} \quad (8)$$

for $k \in \{\rightarrow, \leftarrow, \nleftrightarrow\}$ and $1 \leq j \leq R$.

In summary, the EM-based accuracy estimation algorithm works as follows:

(i) Take initial estimates of the parameters $\boldsymbol{p}, \boldsymbol{\gamma}$.
(ii) Use equation (7) and the current estimates of the parameters to calculate estimates of the expectation of the hidden variables g_i.
(iii) Use equations (8) to obtain new estimates of the parameters.
(iv) Repeat steps (ii) and (iii) until the results converge.

The EM algorithm yields only local optima, but the considerable experience with the algorithm indicates that the results are usually satisfactory.

5 Marginalization-Based Score

The scoring approach proposed in the previous section consists of two steps. In the first step, the experts' accuracies are estimated, then in the second step, the estimated accuracies are used to score the structures. Obviously, the reliability of this score depends on the reliability of the estimated accuracies in the first step. When we are not confident about the estimated accuracies, this approach is not appropriate. In this section, we introduce an alternative approach that is based on marginalizing out the accuracy parameters instead of explicitly estimating them.

Since the estimated experts' accuracies are not explicitly used in the marginalization-based score, only the data D and experts' opinions O are used for scoring a candidate

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

structure G. The posterior probability of G given D and O is a reasonable measure for this purpose:

$$P(G \mid D, O) \propto P(G, D, O) = P(G)\, P(D \mid G)\, P(O \mid G, D).$$

Based on the statement 4 in Table 1, $P(O \mid G, D)$ is simplified as $P(O \mid G)$. Therefore, we define our marginalization-based score as:

$$\text{Score}_{marg}(G; D, O) = \log P(G) + \log P(D \mid G) + \log P(O \mid G).$$

Comparing the above scoring function with the explicit-accuracy-based score (1), the only difference is the last term. In the rest of this section, we explain how to compute $\log P(O \mid G)$ to complete the computation of the marginalization-based score.

According to statement 6 in Table 1,

$$\log P(O \mid G) = \sum_{i=1}^{R} \log P(O^i \mid G). \tag{9}$$

To compute $P(O^i \mid G)$, we marginalize out the accuracy parameters:

$$P(O^i \mid G) = \int_0^1 \int_0^{1-\gamma_1^i} \int_0^1 \Big(P(\gamma_1^i, \gamma_2^i, \gamma_3^i \mid G) \times P(O^i \mid \gamma_1^i, \gamma_2^i, \gamma_3^i, G) \Big)\, d\gamma_3^i\, d\gamma_2^i\, d\gamma_1^i. \tag{10}$$

Note that the domain of integration for γ_2^i is not $[0, 1]$ but is $[0, 1 - \gamma_1^i]$, because $\gamma_1^i + \gamma_2^i$ must be lower than or equal to 1.

According to statements 1 and 14 in Table 1,

$$P(\gamma_1^i, \gamma_2^i, \gamma_3^i \mid G) = P(\gamma_1^i, \gamma_2^i)\, P(\gamma_3^i). \tag{11}$$

In addition, based on statement 9 we have

$$P(O^i \mid \gamma_1^i, \gamma_2^i, \gamma_3^i, G) = \prod_{j=1}^{N} P(O_j^i \mid \gamma_1^i, \gamma_2^i, \gamma_3^i, G),$$

which can be written as

$$P(O^i \mid \gamma_1^i, \gamma_2^i, \gamma_3^i, G) = (\gamma_1^i)^{n_{i1}} (\gamma_2^i)^{n_{i2}} (1 - \gamma_1^i - \gamma_2^i)^{n_{i3}} (\gamma_3^i)^{m_{i1}} \left(\frac{1 - \gamma_3^i}{2} \right)^{m_{i2}}, \tag{12}$$

where

- n_{i1} is the number of existing edges in G that are detected by expert i with correct directions,
- n_{i2} is the number of existing edges in G that are detected by expert i with reverse directions,
- n_{i3} is the number of existing edges in G that are mentioned as absent edges by expert i,
- m_{i1} is the number of absent edges in G that are correctly detected by expert i,
- m_{i2} is the number of absent edges in G that are mentioned as existing edges by expert i.

Plugging equations (11) and (12) into equation (10), we get

$$P(O^i \mid G) = \left(\int_0^1 P(\gamma_3^i)\, (\gamma_3^i)^{m_{i1}} \left(\frac{1 - \gamma_3^i}{2} \right)^{m_{i2}} d\gamma_3^i \right) \times \left(\int_0^1 \int_0^{1-\gamma_1^i} P(\gamma_1^i, \gamma_2^i)\, (\gamma_1^i)^{n_{i1}} (\gamma_2^i)^{n_{i2}} (1 - \gamma_1^i - \gamma_2^i)^{n_{i3}}\, d\gamma_2^i\, d\gamma_1^i \right). \tag{13}$$

We denote the components of the above equation by I_1^i and I_2^i, respectively:

$$I_1^i = \int_0^1 P(\gamma_3^i)\, (\gamma_3^i)^{m_{i1}} \left(\frac{1 - \gamma_3^i}{2} \right)^{m_{i2}} d\gamma_3^i,$$

$$I_2^i = \int_0^1 \int_0^{1-\gamma_1^i} \Big(P(\gamma_1^i, \gamma_2^i)\, (\gamma_1^i)^{n_{i1}} (\gamma_2^i)^{n_{i2}} (1 - \gamma_1^i - \gamma_2^i)^{n_{i3}} \Big)\, d\gamma_2^i\, d\gamma_1^i.$$

An appropriate distribution for $P(\gamma_3^i)$ in I_1^i is the Beta distribution. If the shape parameters of this distribution are denoted by β_{i1} and β_{i2}, we have

$$P(\gamma_3^i) = \frac{1}{B(\beta_{i1}, \beta_{i2})} (\gamma_3^i)^{\beta_{i1} - 1} (1 - \gamma_3^i)^{\beta_{i2} - 1}, \tag{14}$$

where

$$B(\beta_{i1}, \beta_{i2}) = \int_0^1 t^{\beta_{i1} - 1} (1 - t)^{\beta_{i2} - 1} dt \tag{15}$$

is the Beta function. Note that the shape parameters can be different for different experts. Nevertheless, we use the same parameters for all experts in our experiments.

Plugging equation (14) into the definition of I_1^i, we have

$$I_1^i = \frac{\int_0^1 (\gamma_3^i)^{\beta_{i1} + m_{i1} - 1} (1 - \gamma_3^i)^{\beta_{i2} + m_{i2} - 1}\, d\gamma_3^i}{2^{m_{i2}} \times B(\beta_{i1}, \beta_{i2})},$$

which can be written as

$$I_1^i = \frac{B(\beta_{i1} + m_{i1}, \beta_{i2} + m_{i2})}{2^{m_{i2}} \times B(\beta_{i1}, \beta_{i2})}. \tag{16}$$

After deriving a closed-form formula for I_1^i, we now turn our attention to I_2^i. We use a Dirichlet distribution $\text{Dir}(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ for $P(\gamma_1^i, \gamma_2^i)$:

$$P(\gamma_1^i, \gamma_2^i) = \frac{(\gamma_1^i)^{\alpha_{i1} - 1} (\gamma_2^i)^{\alpha_{i2} - 1} (1 - \gamma_1^i - \gamma_2^i)^{\alpha_{i3} - 1}}{B(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})},$$

where $B(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})$ is the multivariate Beta function. Using this prior in the definition of I_2^i we have

$$I_2^i = \frac{1}{B(\alpha_{i1}, \alpha_{i2}, \alpha_{i3})} \int_0^1 (\gamma_1^i)^{\alpha_{i1} + n_{i1} - 1} \times \left[\int_0^{1-\gamma_1^i} (\gamma_2^i)^{\alpha_{i2} + n_{i2} - 1} (1 - \gamma_1^i - \gamma_2^i)^{\alpha_{i3} + n_{i3} - 1}\, d\gamma_2^i \right] d\gamma_1^i.$$

Changing the variable t in integral (15) to γ_2^i by substituting $t = \frac{\gamma_2^i}{1 - \gamma_1^i}$, the inner integral in the above equation is

$$\int_0^{1-\gamma_1^i} (\gamma_2^i)^{\alpha_{i2} + n_{i2} - 1} (1 - \gamma_1^i - \gamma_2^i)^{\alpha_{i3} + n_{i3} - 1}\, d\gamma_2^i = B(\alpha_{i2} + n_{i2}, \alpha_{i3} + n_{i3}) \times (1 - \gamma_1^i)^{\alpha_{i2} + n_{i2} + \alpha_{i3} + n_{i3} - 1},$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 2
Description of the Networks Used in the Simulation Experiments

Name	Description	Nodes	Edges
Asia	Diagnosing some respiratory diseases	8	8
Insurance	Evaluating car insurance risks	27	52
Alarm	Monitoring patients in intensive care	37	46
Hailfinder	Predicting summer hails in northern Colorado	56	66

and therefore,

$$I_2^i = \frac{B(\alpha_{i2}+n_{i2},\alpha_{i3}+n_{i3})}{B(\alpha_{i1},\alpha_{i2},\alpha_{i3})} \times \int_0^1 (\gamma_1^i)^{\alpha_{i1}+n_{i1}-1}(1-\gamma_1^i)^{\alpha_{i2}+n_{i2}+\alpha_{i3}+n_{i3}-1}\, d\gamma_1^i,$$

which can be written as

$$I_2^i = \frac{1}{B(\alpha_{i1},\alpha_{i2},\alpha_{i3})} \times B(\alpha_{i2}+n_{i2},\alpha_{i3}+n_{i3}) \times B(\alpha_{i1}+n_{i1},\alpha_{i2}+n_{i2}+\alpha_{i3}+n_{i3}). \quad (17)$$

Based on equation (13), we can compute $P(O^i \mid G)$ by multiplying equations (16), (17). So,

$$\begin{aligned}\log P(O^i \mid G) =& \log B(\beta_{i1}+m_{i1},\beta_{i2}+m_{i2}) \\ & -m_{i2}\log 2 - \log B(\beta_{i1},\beta_{i2}) \\ & + \log B(\alpha_{i1}+n_{i1},\alpha_{i2}+n_{i2}+\alpha_{i3}+n_{i3}) \\ & + \log B(\alpha_{i2}+n_{i2},\alpha_{i3}+n_{i3}) \\ & - \log B(\alpha_{i1},\alpha_{i2},\alpha_{i3}).\end{aligned}$$

Finally, based on equation (9) we get $\log P(O \mid G)$ by summing the above expression for $i = 1,\ldots,R$, which completes the computation of the marginalization-based score.

6 Experiments

The developed scores are evaluated in this section using simulated experts (subsection 6.1) and real experts (subsection 6.2).

6.1 Simulation Experiments

To evaluate the developed scores, some of the experiments are performed on simulated experts. The merit of simulation is that we can change the values of different parameters, such as the experts' accuracies or the amount of available knowledge, and evaluate the scores under different conditions. In this part of the paper, we present the setup of our simulation experiments and discuss the obtained results.

6.1.1 Experimental Setup

We use four Bayesian networks which have been widely used in the structure learning experiments: Asia [31], Insurance [32], Alarm [33], and Hailfinder [34], briefly described in Table 2.

Our experiments are implemented in a MATLAB environment using the Bayes net toolbox [35] and the BNT structure learning package [36]. For each network, we generate the data samples and experts' opinions and learn the structure using different scoring functions. Comparing the learned network with the gold-standard structure reveals the effectiveness of the corresponding scoring function. In addition to comparing the DAGs, we also compare the CPDAGs representing the equivalence classes of the learned structure and the gold-standard network [21]. The reason for comparing the CPDAGs is that we do not penalize for structural differences that cannot be distinguished only based on data [37].

TABLE 3
The Accuracy Parameters Assigned to Experts in Simulated Populations

	Weak			Mediocre			Good		
	γ_1	γ_2	γ_3	γ_1	γ_2	γ_3	γ_1	γ_2	γ_3
1	0.15	0.80	0.85	0.15	0.80	0.85	0.15	0.80	0.85
2	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30
3	0.75	0.10	0.90	0.75	0.10	0.90	0.75	0.10	0.90
4	0.40	0.25	0.50	0.40	0.25	0.50	0.85	0.05	0.85
5	0.45	0.35	0.45	0.45	0.35	0.45	0.70	0.15	0.80
6	0.55	0.20	0.60	0.55	0.20	0.60	0.75	0.15	0.70
7	0.20	0.15	0.50	0.20	0.15	0.95	0.20	0.15	0.95
8	0.33	0.33	0.33	0.90	0.05	0.80	0.90	0.05	0.80
9	0.50	0.30	0.40	0.70	0.20	0.70	0.70	0.20	0.70
10	0.30	0.50	0.30	0.60	0.30	0.65	0.80	0.10	0.90
Mean	0.39	0.33	0.51	0.50	0.27	0.67	0.61	0.21	0.78

There may be three types of errors in the learned DAG (CPDAG):

- Wrong Connection where an absent edge in the original graph is available in the learned network,
- Missed Edge where an available edge in the original graph is missed in the learned structure,
- Wrong Orientation where one edge has different orientations in the original graph and the learned structure. Note that, when comparing two CPDAGs such as G_1 and G_2, if one edge is undirected in G_1 and directed in G_2, this is also considered as wrong orientation error, since there is at least one graph in the equivalence class of G_1 where its corresponding edge has wrong orientation than that of G_2.

The total number of these errors is called the structural Hamming distance (SHD) [37], [38].

In our simulations, we consider three different populations each with $R = 10$ experts. According to the experts' accuracies, these populations are labeled as "weak", "mediocre" and "good". Table 3 lists the details of the experts' accuracies in these populations. Three experts are equally accurate in all populations. Three other experts are equally accurate in the "weak" and "mediocre" populations, but more accurate in the "good" population. The next three experts are equally accurate in the "mediocre" and "good" populations, but less accurate in the "weak" population. Note that since higher γ_1 and γ_3 means more accurate experts, these parameters have higher values in the "good" population. On the other hand, since higher γ_2 means less accurate experts, this parameter has higher values in the "weak" population.[1]

In our experiments, we compare six different functions:

1) **Data**: This function neglects the experts' opinions and only uses the data D to score the structures. We use the marginal likelihood part of the BDeu score [2], introduced in Section 2, for this purpose.

1. The detailed description of the process of simulating the experts' opinions based on their accuracy parameters is available at http://ceit.aut.ac.ir/~amirkhani.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 4
The True Probability Distributions Over Edge Types for the Bayesian Networks Used in the Simulation Experiments

BN	$p_{\rightarrow}$	$p_{\leftarrow}$	$p_{\nleftrightarrow}$
Asia	0.11	0.18	0.71
Insurance	0.07	0.08	0.85
Alarm	0.04	0.03	0.93
Hailfinder	0.02	0.02	0.96

2) **Expert**: This function neglects the data D and only uses the experts' opinions O to decide about the Bayesian network structure. It uses a majority voting approach, in which, for each pair (X, Y), the status that the majority of experts agree on is considered as the status of the edge between X and Y. In the case of a tie, a random decision is made.
3) **PE**: Stands for perfect experts, this scoring function assumes that all experts are completely accurate. For this, the explicit-accuracy-based score (1) is used, where γ_1 and γ_3 parameters of all experts are set to one, and γ_2 is zero.
4) **Mean**: This function also exploits both data and experts' opinions using the explicit-accuracy-based score (1). It considers the same accuracies for all experts to resemble the method proposed in [16]. For the accuracy parameters, it uses the mean of true accuracy parameters of all experts in each population. More precisely, if the opinions are generated from the "weak" population, $\gamma_1, \gamma_2, \gamma_3$ are set to 0.4, 0.35, 0.5, respectively. For the "mediocre" population, these parameters are set to 0.5, 0.25, 0.65, respectively. Finally, for the "good" population, 0.6, 0.2, 0.8 are used as the accuracy parameters for all experts. This scoring function is included in the experiments to compare the best achievable results from a method such as [16] that considers the same accuracy levels for all experts with the methods such as the EM-based method proposed in Section 4 which try to estimate the accuracies of all experts.
5) **EM**: In this function, we first estimate the experts' accuracies using the EM-based algorithm introduced in subsection 4.2, and then, score the structures using the explicit-accuracy-based score (1). As the initial prior distribution over edge types, we use $\boldsymbol{p} = \{p_{\rightarrow} = 0.1, p_{\leftarrow} = 0.1, p_{\nleftrightarrow} = 0.8\}$, since we know that most real world Bayesian network structures are sparse. The true distributions for the used Bayesian networks are presented in Table 4. For the initial accuracy parameters $\boldsymbol{\gamma}$, we assume that we have the mean of true accuracy parameters of all experts a priori, and therefore, we initialize $\boldsymbol{\gamma}$ with the values used for the Mean scoring function. We use these initial values because we want to provide the same prior information for both Mean and EM scores, and can fairly compare their results. The EM algorithm stops when the absolute change in the estimated accuracies is smaller than 0.001.
6) **Marg**: This function is the marginalization-based score introduced in Section 5. For the parameters $\beta_{i1}, \beta_{i2}, \alpha_{i1}, \alpha_{i2}, \alpha_{i3}$, we use the same values for all experts, again using the mean of true accuracy parameters. If the mean of true accuracy parameters of all experts in a particular population is $\bar{\gamma}_1, \bar{\gamma}_2, \bar{\gamma}_3$, we have

$$\beta_{i1} = c\bar{\gamma}_3, \beta i2 = c(1 - \bar{\gamma}_3),$$
$$\alpha_{i1} = c\bar{\gamma}_1, \alpha_{i2} = c\bar{\gamma}_2, \alpha_{i3} = c(1 - \bar{\gamma}_1 - \bar{\gamma}_2), \quad (18)$$

for $i = 1, \ldots, R$, where c is a constant coefficient. In the following, wherever the value of c is not clearly mentioned, its value is 10. At the end of this section, we evaluate the influence of this coefficient and show that its value does not have such a considerable impact on the obtained results.

In all of the above scoring functions, for the log-likelihood $\log P(D \mid G)$, the corresponding component from the BDeu score is used, and the parameter representing the equivalent sample size is set to 1. For the prior distribution over structures $P(G)$, the uniform prior is used. Finally, as the search procedure, we use the same hill-climbing search as [2] introduced in Section 2, starting from an empty network.

The number of opinions is controlled by a parameter $\nu \in [0, 1]$. If there are N pairs of variables and R experts, the total number of opinions provided by all experts is equal to $\text{round}(\nu \times R \times N)$, where the function $\text{round}(x)$ outputs the closest integer to x. In our experiments, the value of the parameter ν is selected from $\{0.3, 0.4, 0.5, 0.6\}$.

For the training dataset D, two different sizes are considered: 1000 and 5000. To reduce the effect of randomness in the reported results, we repeat each experiment 10 times and report the average results over these iterations. In other words, for each Bayesian network, we generate 10 different datasets with 1000 samples and 10 different datasets with 5000 samples; and for each triple {Bayesian network, population, ν}, we simulate 10 different opinion sets. Then, in each experiment, we use one dataset and one opinion set to learn the Bayesian network structure.

6.1.2 Results and Discussion

Tables 5 to 8 show the obtained structural Hamming distances for the Asia, Insurance, Alarm, and Hailfinder networks, respectively. Each table includes three subtables, one for each population. In each row, the best obtained results are indicated by boldface values.

According to these tables, we observe that:

- The results obtained for the *Expert* scenario show that using only the experts' opinions gives rise to very low-accurate networks, especially for large networks and *weak* populations. The main reason is that, in our experiments in accordance with the real world, the experts are not forced to present their opinions regarding all parts of the network, and we do not have enough information to accurately decide about the parts of the network that the majority of the experts are silent about.
- For the *weak* populations, the second worst results (after the *Expert* scenario) are related to the *PE* scenario. The reason is that when there are considerable errors in the provided experts' opinions, the raw usage of these opinions (as in the *PE* scenario) reduces the accuracy of the learned network.
- The third worst results for the *weak* populations are related to the *EM-based* score. It is obvious that for the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

scores which try to explicitly use the estimated experts' accuracies, if the estimation process is not successful, it is not so useful to use the experts' knowledge based on the explicit-accuracy-based score. This is the case for the *EM-based* scenario for the *weak* populations. In fact, it is well known that the EM algorithm depends on the starting point and may converge to a local optimum. On the other hand, for the *weak* populations, the starting points are really questionable (which is why we call them *weak*). Therefore, the *EM-based* method fails to obtain robust results for the *weak* populations.

- In the majority of cases for the *mediocre* and *good* populations, the *EM-based* method obtains either the best or the second best results. This is because, in these situations, the EM algorithm starts with good enough starting points. The conclusion is that, when we are faced with accurate enough experts' opinions, the *EM-based* method obtains reasonable results. It may be asked, what if we do not know anything about the accuracy level of the opinions at hand? The *EM-based* method may get stuck in a local optimum far from true experts' accuracies. This was the main stimulus for proposing an alternative approach, the *marginalization-based* scoring function, in Section 5. It is clear from the tables that the *marginalization-based* score obtains more robust results.
- In the majority of cases for the *weak* populations, the *marginalization-based* score yields better results than the other scores. On the other hand, in situations where the *marginalization-based* score is not the winner, it obtains acceptable results compared to the best achieved results. Therefore, if we have no idea about the accuracy levels of our experts, it is better to use the *marginalization-based* score.
- In the majority of cases for the *mediocre* and *good* populations, the results obtained by the *EM-based* score are superior to those obtained by the *Mean* score. This shows that if we can successfully estimate the experts' accuracies, using these estimated values for different experts is better than considering a constant accuracy level for all experts.
- One may think that with larger training datasets (e.g., $|D| = 5000$ in comparison with $|D| = 1000$), the structural Hamming distances will become smaller. However, this is not always the case. It can be explained by noting that learning Bayesian networks aims at two goals: (1) to obtain the structure of the domain, and (2) to obtain the probability distribution underlying the domain. These two goals are sometimes in conflict: adding an extra edge to a Bayesian network may increase the structural Hamming distance, yet resulting in a more accurate probability distribution. This is sometimes the case with larger datasets, where complex networks with more wrong connections are preferred over the true network, yielding a more accurate probability distribution. There is some literature that supports this claim [16], [39]. In these situations, the experts' opinions can be really useful to obtain more accurate structures of the domain, such as for the Hailfinder network in the Table 8.

TABLE 5
The Obtained Structural Hamming Distances for the Asia Network

(a) Weak population

$\|D\|$	ν	DAG Data	DAG Expert	DAG PE	DAG Mean	DAG EM	DAG Marg	CPDAG Data	CPDAG Expert	CPDAG PE	CPDAG Mean	CPDAG EM	CPDAG Marg
1000	0.3	7.2	13.6	12.0	6.8	12.7	**6.7**	7.0	14.4	12.3	**6.2**	13.5	6.5
	0.4	7.2	11.2	11.0	6.0	8.7	**5.7**	7.0	12.3	12.1	5.9	9.0	**5.3**
	0.5	7.2	11.1	9.4	5.0	9.2	**3.8**	7.0	12.2	10.2	5.2	10.3	**4.1**
	0.6	7.2	12.0	8.8	4.3	7.3	**3.9**	7.0	13.7	10.0	**4.4**	8.4	4.5
	Avg	7.2	12.0	10.3	5.5	9.5	**5.0**	7.0	13.2	11.2	5.4	10.3	**5.1**
5000	0.3	9.3	13.6	10.5	**6.9**	10.3	7.4	9.5	14.4	10.9	**7.0**	11.0	7.7
	0.4	9.3	11.2	9.2	5.4	7.6	**4.9**	9.5	12.3	9.9	4.6	7.6	**4.5**
	0.5	9.3	11.1	9.3	5.2	9.2	**3.8**	9.5	12.2	10.2	5.1	9.9	**3.9**
	0.6	9.3	12.0	8.8	4.5	7.1	**4.1**	9.5	13.7	10.0	4.1	7.6	**3.4**
	Avg	9.3	12.0	9.5	5.5	8.5	**5.1**	9.5	13.2	10.3	5.2	9.0	**4.9**

(b) Mediocre population

$\|D\|$	ν	DAG Data	DAG Expert	DAG PE	DAG Mean	DAG EM	DAG Marg	CPDAG Data	CPDAG Expert	CPDAG PE	CPDAG Mean	CPDAG EM	CPDAG Marg
1000	0.3	7.2	8.6	4.7	3.9	5.0	**2.8**	7.0	9.9	5.3	3.8	5.0	**2.6**
	0.4	7.2	7.6	5.2	5.1	5.5	**4.3**	7.0	8.6	5.6	4.7	5.5	**4.4**
	0.5	7.2	5.1	3.0	2.8	3.3	**1.7**	7.0	6.4	2.9	2.1	3.2	**1.1**
	0.6	7.2	4.4	2.7	2.1	2.8	**1.3**	7.0	5.8	3.7	1.9	3.1	**1.2**
	Avg	7.2	6.4	3.9	3.5	4.2	**2.5**	7.0	7.7	4.4	3.1	4.2	**2.3**
5000	0.3	9.3	8.6	4.6	3.4	5.1	**2.6**	9.5	9.9	4.7	2.8	4.8	**2.0**
	0.4	9.3	7.6	5.4	4.7	**4.4**	4.7	9.5	8.6	5.6	**4.0**	4.3	4.2
	0.5	9.3	5.1	2.9	2.5	3.6	**1.8**	9.5	6.4	2.6	2.7	3.6	**1.9**
	0.6	9.3	4.4	2.7	3.1	2.2	**2.1**	9.5	5.8	3.1	2.6	2.3	**2.0**
	Avg	9.3	6.4	3.9	3.4	3.8	**2.8**	9.5	7.7	4.0	3.0	3.8	**2.5**

(c) Good population

$\|D\|$	ν	DAG Data	DAG Expert	DAG PE	DAG Mean	DAG EM	DAG Marg	CPDAG Data	CPDAG Expert	CPDAG PE	CPDAG Mean	CPDAG EM	CPDAG Marg
1000	0.3	7.2	5.7	4.2	4.4	3.9	**3.2**	7.0	7.6	4.9	4.3	4.4	**2.6**
	0.4	7.2	4.6	**2.0**	3.3	2.6	**2.0**	7.0	5.4	2.3	3.1	2.7	**1.8**
	0.5	7.2	3.7	1.2	2.2	**0.9**	1.1	7.0	5.1	1.4	2.2	1.2	**1.1**
	0.6	7.2	2.4	1.1	**0.9**	**0.9**	**0.9**	7.0	4.0	1.1	**0.4**	0.7	0.7
	Avg	7.2	4.1	2.1	2.7	2.1	**1.8**	7.0	5.5	2.4	2.5	2.3	**1.6**
5000	0.3	9.3	5.7	3.9	3.9	4.2	**3.5**	9.5	7.6	4.1	3.5	4.4	**2.8**
	0.4	9.3	4.6	2.8	4.1	**2.2**	2.7	9.5	5.4	3.0	3.6	**2.0**	2.3
	0.5	9.3	3.7	1.4	1.2	0.8	**0.3**	9.5	5.1	1.3	0.7	1.1	**0.1**
	0.6	9.3	2.4	1.1	1.2	1.1	**0.9**	9.5	4.0	0.7	0.6	1.0	**0.5**
	Avg	9.3	4.1	2.3	2.6	2.1	**1.9**	9.5	5.5	2.3	2.1	2.1	**1.4**

TABLE 6
The Obtained Structural Hamming Distances for the Insurance Network

(a) Weak population

$\|D\|$	ν	DAG Data	DAG Expert	DAG PE	DAG Mean	DAG EM	DAG Marg	CPDAG Data	CPDAG Expert	CPDAG PE	CPDAG Mean	CPDAG EM	CPDAG Marg
1000	0.3	29.2	126.9	53.9	29.6	36.9	**28.0**	36.6	132.0	61.6	36.8	43.1	**36.0**
	0.4	29.2	107.0	49.9	28.0	29.1	**23.3**	36.6	114.5	58.1	36.1	34.7	**28.9**
	0.5	29.2	108.2	49.5	31.1	**28.6**	29.0	36.6	115.0	56.6	39.4	36.6	**34.7**
	0.6	29.2	104.0	50.3	30.0	30.6	**25.7**	36.6	110.2	57.2	36.6	39.1	**34.8**
	Avg	29.2	111.5	50.9	29.7	31.3	**26.5**	36.6	117.9	58.4	37.2	38.4	**35.6**
5000	0.3	**24.8**	126.9	41.8	27.4	33.9	28.6	**22.8**	132.0	51.7	30.2	41.6	30.4
	0.4	**24.8**	107.0	37.7	26.0	27.1	27.3	**22.8**	114.5	47.0	29.9	34.2	31.2
	0.5	**24.8**	108.2	39.8	28.7	27.6	30.0	**22.8**	115.0	46.7	34.5	32.7	35.3
	0.6	**24.8**	104.0	39.9	28.0	27.1	26.5	**22.8**	110.2	48.2	29.4	31.5	31.9
	Avg	**24.8**	111.5	39.8	27.5	28.9	28.1	**22.8**	117.9	48.4	31.0	35.0	32.2

(b) Mediocre population

$\|D\|$	ν	DAG Data	DAG Expert	DAG PE	DAG Mean	DAG EM	DAG Marg	CPDAG Data	CPDAG Expert	CPDAG PE	CPDAG Mean	CPDAG EM	CPDAG Marg
1000	0.3	29.2	74.0	27.2	**21.0**	21.5	22.8	36.6	82.8	36.1	**26.1**	28.1	32.0
	0.4	29.2	63.3	23.6	21.9	20.6	**18.8**	36.6	72.2	31.5	32.1	28.4	**26.5**
	0.5	29.2	56.0	**20.9**	24.6	22.8	22.7	36.6	65.1	**24.8**	31.5	31.3	28.7
	0.6	29.2	49.5	21.8	26.2	**16.7**	23.2	36.6	58.4	30.3	32.7	**24.5**	30.0
	Avg	29.2	60.7	23.4	23.4	**20.4**	21.9	36.6	69.6	30.7	30.6	**28.1**	29.3
5000	0.3	24.8	74.0	21.0	**19.1**	21.5	19.6	22.8	82.8	28.5	**21.1**	25.0	22.4
	0.4	24.8	63.3	22.3	24.1	24.5	**20.9**	**22.8**	72.2	29.2	28.7	26.4	24.9
	0.5	24.8	56.0	**20.6**	26.7	21.6	20.7	**22.8**	65.1	26.2	30.2	26.1	24.6
	0.6	24.8	49.5	21.0	23.7	**16.1**	19.7	22.8	58.4	27.7	27.0	**19.2**	24.8
	Avg	24.8	60.7	21.2	23.4	20.9	**20.2**	**22.8**	69.6	27.9	26.8	24.2	24.2

(c) Good population

$\|D\|$	ν	DAG Data	DAG Expert	DAG PE	DAG Mean	DAG EM	DAG Marg	CPDAG Data	CPDAG Expert	CPDAG PE	CPDAG Mean	CPDAG EM	CPDAG Marg
1000	0.3	29.2	57.7	**18.3**	21.9	18.7	19.4	36.6	67.9	25.6	30.7	**24.5**	26.1
	0.4	29.2	43.5	**13.6**	19.2	14.9	17.3	36.6	54.3	**21.2**	27.3	21.9	26.3
	0.5	29.2	28.7	**7.9**	21.1	14.3	19.0	36.6	41.1	**11.9**	27.7	21.3	25.4
	0.6	29.2	20.4	**5.5**	16.2	12.6	13.9	36.6	29.9	**7.9**	23.8	19.0	19.6
	Avg	29.2	37.6	**11.3**	19.6	15.1	17.4	36.6	48.3	**16.6**	27.4	21.7	24.4
5000	0.3	24.8	57.7	19.7	22.4	**16.9**	19.4	22.8	67.9	27.1	27.0	**20.9**	22.3
	0.4	24.8	43.5	14.0	18.5	**12.8**	18.6	22.8	54.3	18.8	18.8	**14.9**	20.2
	0.5	24.8	28.7	12.7	19.1	**9.7**	17.4	22.8	41.1	18.0	24.4	**12.9**	21.5
	0.6	24.8	20.4	**8.2**	16.3	9.4	13.8	22.8	29.9	11.5	21.3	**10.0**	15.6
	Avg	24.8	37.6	13.7	19.1	**12.2**	17.3	22.8	48.3	18.9	22.9	**14.7**	19.9

As our final simulation experiment, we investigate the impact of coefficient c in equation (18) on the accuracy of the marginalization-based score. We report the obtained

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 7
The Obtained Structural Hamming Distances for the Alarm Network

(a) Weak population

		DAG						CPDAG					
$\|D\|$	ν	Data	Expert	PE	Mean	EM	Marg	Data	Expert	PE	Mean	EM	Marg
1000	0.3	30.2	212.7	108.6	31.2	44.7	**28.9**	31.9	213.7	110.5	34.8	47.5	**31.3**
	0.4	30.2	186.6	98.7	34.3	34.0	**28.2**	**31.9**	188.2	100.3	37.5	38.9	32.1
	0.5	30.2	173.2	93.1	31.2	33.0	**28.1**	31.9	174.3	95.1	32.2	36.2	**31.1**
	0.6	30.2	166.5	94.0	29.4	28.9	**26.0**	31.9	167.8	96.0	**31.1**	33.5	31.2
	Avg	30.2	184.8	98.6	31.5	35.1	**27.8**	31.9	186.0	100.5	33.9	39.0	**31.4**
5000	0.3	29.5	212.7	83.8	29.9	38.4	**28.4**	**30.5**	213.7	87.0	33.2	41.3	**30.5**
	0.4	29.5	186.6	79.9	31.1	31.5	**27.8**	**30.5**	188.2	83.0	34.0	34.9	**30.5**
	0.5	29.5	173.2	77.0	30.2	32.6	**24.8**	30.5	174.3	78.9	30.9	35.0	**28.3**
	0.6	29.5	166.5	76.8	27.0	26.4	**24.8**	30.5	167.8	78.8	29.6	29.0	**28.0**
	Avg	29.5	184.8	79.4	29.6	32.2	**26.5**	30.5	186.0	81.9	31.9	35.0	**29.3**

(b) Mediocre population

		DAG						CPDAG					
$\|D\|$	ν	Data	Expert	PE	Mean	EM	Marg	Data	Expert	PE	Mean	EM	Marg
1000	0.3	30.2	123.0	61.8	29.1	30.5	**25.3**	31.9	125.5	65.7	32.7	34.8	**29.4**
	0.4	30.2	95.9	48.6	28.0	**21.7**	23.4	31.9	98.7	51.8	30.6	**25.3**	26.0
	0.5	30.2	87.0	43.7	29.9	**21.9**	28.1	31.9	89.3	47.0	33.0	**24.3**	32.2
	0.6	30.2	78.5	38.1	22.5	**15.6**	17.6	31.9	81.1	40.1	26.8	**17.8**	20.3
	Avg	30.2	96.1	48.1	27.4	**22.4**	23.6	31.9	98.7	51.1	30.8	**25.5**	27.0
5000	0.3	29.5	123.0	52.5	27.7	26.9	**26.2**	**30.5**	125.5	56.8	**30.5**	**30.5**	31.0
	0.4	29.5	95.9	42.1	28.5	**20.5**	22.0	30.5	98.7	46.5	31.6	**23.1**	24.5
	0.5	29.5	87.0	41.9	28.8	**24.2**	29.8	30.5	89.3	45.6	31.5	**26.8**	33.1
	0.6	29.5	78.5	36.9	22.3	**17.2**	19.5	30.5	81.1	40.3	25.2	**19.1**	22.0
	Avg	29.5	96.1	43.4	26.8	**22.2**	24.4	30.5	98.7	47.3	29.7	**24.9**	27.7

(c) Good population

		DAG						CPDAG					
$\|D\|$	ν	Data	Expert	PE	Mean	EM	Marg	Data	Expert	PE	Mean	EM	Marg
1000	0.3	30.2	87.9	38.3	21.7	**17.6**	**17.6**	31.9	90.6	42.4	25.8	**21.3**	21.5
	0.4	30.2	62.4	26.5	20.8	**16.1**	18.0	31.9	65.4	28.7	25.1	**19.6**	21.1
	0.5	30.2	37.7	13.3	16.3	**8.2**	10.6	31.9	40.8	15.0	19.6	**10.0**	12.1
	0.6	30.2	36.1	15.1	15.7	**8.8**	9.0	31.9	39.8	17.3	18.8	**9.6**	11.2
	Avg	30.2	56.0	23.3	18.6	**12.7**	13.8	31.9	59.2	25.8	22.3	**15.1**	16.5
5000	0.3	29.5	87.9	33.4	20.2	15.5	**14.6**	30.5	90.6	37.5	24.0	**18.3**	**18.3**
	0.4	29.5	62.4	26.2	19.8	**16.8**	17.2	30.5	65.4	29.2	23.3	**20.7**	21.1
	0.5	29.5	37.7	13.3	11.5	**8.1**	9.1	30.5	40.8	16.6	14.2	**9.7**	10.8
	0.6	29.5	36.1	15.6	15.5	**7.4**	10.4	30.5	39.8	18.6	18.4	**8.9**	12.7
	Avg	29.5	56.0	22.1	16.8	**11.9**	12.8	30.5	59.2	25.5	20.0	**14.4**	15.7

TABLE 8
The Obtained Structural Hamming Distances for the Hailfinder Network

(a) Weak population

		DAG						CPDAG					
$\|D\|$	ν	Data	Expert	PE	Mean	EM	Marg	Data	Expert	PE	Mean	EM	Marg
1000	0.3	67.0	409.5	79.1	40.6	48.0	**35.0**	70.1	415.9	84.4	45.5	52.5	**40.5**
	0.4	67.0	399.9	86.4	44.2	45.1	**36.3**	70.1	405.5	92.3	48.9	49.7	**41.5**
	0.5	67.0	375.3	85.5	45.0	45.0	**44.3**	70.1	380.9	92.2	48.9	49.2	**48.7**
	0.6	67.0	367.6	90.3	44.2	46.1	**36.1**	70.1	374.1	96.7	49.4	51.3	**41.5**
	Avg	67.0	388.1	85.3	43.5	46.0	**37.9**	70.1	394.1	91.4	48.2	50.7	**43.1**
5000	0.3	69.4	409.5	65.2	36.3	45.6	**28.9**	71.2	415.9	70.5	37.9	48.5	**30.3**
	0.4	69.4	399.9	71.1	40.4	42.8	**30.4**	71.2	405.5	74.8	42.3	44.2	**31.5**
	0.5	69.4	375.3	72.9	42.5	40.9	**39.3**	71.2	380.9	76.1	44.3	41.8	**41.5**
	0.6	69.4	367.6	71.5	41.2	38.7	**30.9**	71.2	374.1	77.6	44.8	40.0	**32.8**
	Avg	69.4	388.1	70.2	40.1	42.0	**32.4**	71.2	394.1	74.8	42.3	43.6	**34.0**

(b) Mediocre population

		DAG						CPDAG					
$\|D\|$	ν	Data	Expert	PE	Mean	EM	Marg	Data	Expert	PE	Mean	EM	Marg
1000	0.3	67.0	282.6	57.4	37.2	34.3	**30.0**	70.1	291.3	62.1	41.5	38.7	**35.6**
	0.4	67.0	196.6	43.0	33.4	31.3	**27.5**	70.1	206.3	47.4	39.5	37.6	**33.1**
	0.5	67.0	191.2	45.3	31.4	27.7	**26.9**	70.1	200.0	48.5	36.1	33.2	**32.8**
	0.6	67.0	158.1	39.0	32.2	**29.5**	30.5	70.1	168.3	46.2	38.6	**36.1**	36.7
	Avg	67.0	207.1	46.2	33.6	30.7	**28.7**	70.1	216.5	51.1	38.9	36.4	**34.6**
5000	0.3	69.4	282.6	50.8	31.6	27.4	**22.1**	71.2	291.3	55.4	34.8	28.6	**23.2**
	0.4	69.4	196.6	36.0	26.4	23.1	**16.2**	71.2	206.3	39.4	28.6	24.8	**17.7**
	0.5	69.4	191.2	37.0	24.0	16.3	**16.2**	71.2	200.0	43.1	25.9	**17.2**	18.3
	0.6	69.4	158.1	33.7	24.5	22.7	**21.3**	71.2	168.3	35.6	26.5	24.9	**23.3**
	Avg	69.4	207.1	39.4	26.6	22.4	**19.0**	71.2	216.5	43.4	29.0	23.9	**20.6**

(c) Good population

		DAG						CPDAG					
$\|D\|$	ν	Data	Expert	PE	Mean	EM	Marg	Data	Expert	PE	Mean	EM	Marg
1000	0.3	67.0	169.7	34.7	30.2	**26.9**	27.7	70.1	180.0	38.8	36.0	**32.8**	33.6
	0.4	67.0	150.3	32.7	29.3	**25.6**	27.4	70.1	161.2	38.0	34.3	**31.1**	32.9
	0.5	67.0	95.2	**22.7**	27.5	23.4	24.2	70.1	108.4	**27.3**	33.7	29.5	30.3
	0.6	67.0	63.2	**17.1**	24.4	21.7	22.5	70.1	75.9	**21.5**	30.6	27.8	28.6
	Avg	67.0	119.6	26.8	27.9	**24.4**	25.5	70.1	131.4	31.4	33.7	**30.3**	31.4
5000	0.3	69.4	169.7	30.7	22.1	19.2	**18.3**	71.2	180.0	32.7	24.5	20.7	**20.5**
	0.4	69.4	150.3	26.7	20.8	**16.1**	18.8	71.2	161.2	29.7	22.6	**17.1**	20.6
	0.5	69.4	95.2	19.2	17.5	13.5	**13.1**	71.2	108.4	21.4	19.6	15.6	**15.4**
	0.6	69.4	63.2	11.7	13.5	**11.5**	11.9	71.2	75.9	12.3	16.0	**13.6**	14.0
	Avg	69.4	119.6	22.1	18.5	**15.1**	15.5	71.2	131.4	24.0	20.7	**16.8**	17.6

results for Asia and Alarm networks as two representative examples. In Tables 5 and 7, the value of the coefficient c for the marginalization-based score is set to 10. Here, we repeat the same experiment for three completely different values from the set $\{0.1, 1, 100\}$. Table 9 shows the obtained structural Hamming distances between the learned structures using the marginalization-based score and the original DAGs. Clearly, the accuracy of the marginalization-based score does not vary substantially when changing the value of c.

TABLE 9
The Obtained Structural Hamming Distances Using the Marginalization-Based Score with Different Values of Coefficient c for Asia and Alarm Bayesian Networks

(a) Weak population

	Asia BN				Alarm BN			
ν/c	0.1	1	10	100	0.1	1	10	100
0.3	7.7	7.5	6.7	6.5	29.4	29.4	28.9	29.0
0.4	5.5	5.3	5.7	6.0	28.7	29.3	28.2	32.4
0.5	4.2	4.0	3.8	5.0	29.5	29.3	28.1	29.3
0.6	4.6	4.3	3.9	4.4	28.9	26.8	26.0	25.8
Avg	5.5	5.3	5.0	5.5	29.1	28.7	27.8	29.1

(b) Mediocre population

	Asia BN				Alarm BN			
ν/c	0.1	1	10	100	0.1	1	10	100
0.3	3.7	3.1	2.8	3.5	24.3	24.9	25.3	27.3
0.4	4.1	4.3	4.3	4.9	22.2	21.3	23.4	24.4
0.5	1.8	1.4	1.7	2.3	28.6	27.6	28.1	27.9
0.6	3.1	2.5	1.3	1.9	22.0	19.0	17.6	21.0
Avg	3.2	2.8	2.5	3.2	24.3	23.2	23.6	25.2

(c) Good population

	Asia BN				Alarm BN			
ν/c	0.1	1	10	100	0.1	1	10	100
0.3	3.9	3.2	3.2	3.2	20.2	18.4	17.6	18.6
0.4	3.4	2.0	2.0	2.3	15.7	16.7	18.0	17.5
0.5	1.4	0.5	1.1	1.4	10.6	10.6	10.6	12.5
0.6	0.9	1.1	0.9	0.6	12.3	8.1	9.0	12.6
Avg	2.4	1.7	1.8	1.9	14.7	13.5	13.8	15.3

6.2 Real World Experiments

To assess whether our methods of expert-opinion-guided structure learning actually work in practice, we have carried out an experiment with real experts. This experiment grew out of our work in computer-aided diagnosis of breast cancer using Bayesian networks [40], [41]. The field of concern was the interpretation of X-ray images by clinical radiologists, in particular X-ray images of breasts, referred to as mammograms. Our previous research has shown that this task can be done by means of a Bayesian network. We expected that the radiologists would be able to draft the structure of a Bayesian network, reflecting their knowledge of mammogram interpretation, depending on their experience with the task. The radiologists had varying amounts of experience in this task: from more than 20 years of specialized experience to no specialized experience at all. Of course, all radiologists have some knowledge of mammogram interpretation.

Eight radiologists were asked to fill in the adjacency matrix shown in Table 10 and seven of them responded to the request.[2] Three of the radiologists were experienced breast

2. The experts' instruction, including the description of the variables, is available at http://ceit.aut.ac.ir/~amirkhani.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 10
Table that the Radiologists had to Fill in as Part of the Experiment. Entries in the Upper Triangular Part of the Table had to be Filled in by $\rightarrow$, $\leftarrow$, $\nleftrightarrow$, or Remain Empty if the Radiologists had no Idea what to Fill in

	Microcalcifications	Spiculation	Location	Age	Lymph Nodes	Skin Retraction	Shape	Size	Breast cancer	Fibrous Tissue Develop	Breast Density	Margin	Nipple Discharge	Architectural Distort	Metastasis	Mass
Microcalcifications																
Spiculation																
Location																
Age																
Lymph Nodes																
Skin Retraction																
Shape																
Size																
Breast cancer																
Fibrous Tissue Develop																
Breast Density																
Margin																
Nipple Discharge																
Architectural Distort																
Metastasis																
Mass																

TABLE 11
The Accuracy Parameters and the Volume of Opinions Provided by Different Experts in the Breast Cancer Experiment

	Accuracy Parameters			Volume of Opinions	
	γ_1	γ_2	γ_3	number	ratio
1	0.17	0.06	0.93	120	1
2	0.44	0.11	0.72	120	1
3	0.50	0.08	0.50	34	0.28
4	0.50	0.17	0.73	120	1
5	0.59	0.24	0.61	104	0.87
6	0.75	0.17	0.63	60	0.5
7	0.91	0.09	0.56	43	0.36
Avg	0.55	0.13	0.67	85.86	0.72

radiologists and also had ample experience as screening radiologists; two were starting breast and screening radiologists, and two were screening radiologists but no breast radiologists. The accuracy parameters and the number and ratio of provided opinions by these experts are presented in Table 11.

In our previous research, we have designed several Bayesian network structures in collaboration with experienced radiologists. These radiologists were different from the radiologists we asked for our current experiment. None of the radiologists had ever seen one of the Bayesian networks we had designed previously. One of those networks is shown in Fig. 6. The Bayesian network model combines clinical features with radiological examination by X-rays. In addition, the Bayesian network integrates the results of microcalcification analysis, which is a separate image analysis procedure.

Training data was generated from the Bayesian network shown in Fig. 6, which is considered as the gold-standard structure.[3] Table 12 shows the obtained structural Hamming distances between the gold-standard structure and the structures learned using the scoring functions mentioned in subsection 6.1.1. In this table, $|D|$ means the number of training data which is selected from the set $\{100, 400, 700, 1000\}$. In order to reduce the effect of randomness in the reported

3. This Bayesian network is available at http://www.cs.ru.nl/~peterl/teaching/CI/networks/bc.net.

TABLE 12
The Obtained Structural Hamming Distances for the Breast Cancer Network with Real Experts

$\|D\|$	Data	Expert	PE	Mean	EM	Marg
100	13.2	32.0	26.4	**9.7**	25.3	10.2
400	8.5	32.0	23.4	**7.2**	19.6	**7.2**
700	6.8	32.0	22.0	**5.8**	18.2	**5.8**
1000	6.8	32.0	21.0	**5.8**	17.9	**5.8**
Avg	8.8	32.0	23.2	**7.1**	20.3	7.3

results, each experiment is repeated 10 times (for 10 different training datasets) and the average results over these repetitions are reported. Finally, the coefficient c in equation (18) for the marginalization-based score is set to 100.

As it is clear from Table 12, the Mean and Marg scoring functions obtained the best results. The success of Mean scoring function is due to the low variance in the experts' accuracies. More precisely, according to Table 11, the accuracies of different experts are not so far from the average values and therefore, using these average values instead of individual accuracies yielded good results.

Although Mean and Marg scores obtained similar results in Table 12, we now show that our marginalization-based scoring function is more reliable. Note that both functions need an estimation of the average experts' accuracies as input. These average values are used as the individual experts' accuracies in the Mean function, and for calculating the parameters $\beta_{i1}, \beta_{i2}, \alpha_{i1}, \alpha_{i2}, \alpha_{i3}$ using equation (18) in the Marg function. Obviously, in real world applications, there might be some errors in the estimated average accuracies. To compare Mean and Marg scoring functions, we studied the behaviors for different levels of errors in this input vector.

Fig. 7 shows the mean and one standard deviation error bars for the structural Hamming distances obtained from Mean and Marg scores. The horizontal axis is the available error in the input average accuracy vector, which is equal to the sum of the absolute errors in the input vector related to the true vector ($[0.55, 0.13, 0.67]$). According to this figure, in general, the marginalization-based score obtains lower structural Hamming distances with lower standard deviations, which shows the robustness of this function compared to the Mean scoring function.

7 Conclusion

This paper focused on exploiting the opinions of multiple domain experts regarding the cause-effect relationships between random variables for structure learning of Bayesian networks. The proposed approach enables structure learning to exploit experts' opinions to learn more accurate network structures than from data alone. Well-known limitations of structure learning algorithms, such as the huge, super-exponential size of the search space and the impossibility to distinguish between Markov-equivalent structures using data alone, motivated this research.

The proposed approach only takes into account realistic assumptions of experts' opinions. For example, experts' opinions need not be error free, and neither have each expert to give a complete judgment of the presence or absence of edges, nor is it necessary that the opinions are conflict free.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2636828, IEEE Transactions on Pattern Analysis and Machine Intelligence

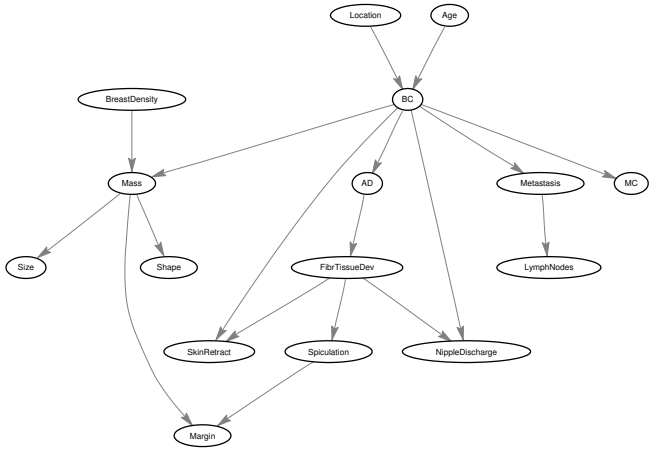


Fig. 6. The structure of the Bayesian network for breast cancer diagnosis.

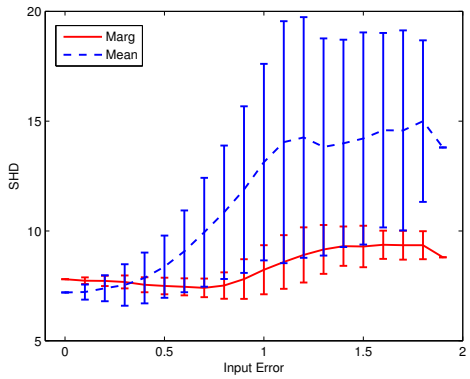


Fig. 7. The mean and one standard deviation error bars for the structural Hamming distances obtained from Mean and Marg scores as functions of the available error in the input average accuracy vector for the breast cancer network with real experts.

To exploit the provided opinions, we introduced two new scoring functions to be used in the score-based Bayesian network structure learning. The main novelty of the proposed scores is that we take into account the natural point of view that different experts have different individual probabilities of correctly labeling the inclusion or exclusion of edges in the structure. The accuracy of each expert was modeled by three parameters. In the first scoring function, the experts' accuracies are first estimated using an expectation-maximization-based algorithm. Then, the estimated values are explicitly used in the scoring process. When we are confident about the estimated accuracies, this scoring function results in robust decisions. On the other hand, when it is not possible to find a confident estimate of experts' accuracies, our second score, the marginalization-based score, which marginalizes out the accuracy parameters results in more robust scores.

Some of the future research directions are (i) to work on relaxing the assumptions made in the development of the EM-based accuracy estimation algorithm described in Section 4.2, (ii) to develop algorithms that use data along with experts' opinions to obtain improved estimates of experts' accuracies, (iii) to use the recently published agreement/disagreement algorithm [42] for estimating the experts' accuracies in the structure learning problem, and (iv) to exploit the experts' opinions for constraint-based structure learning. For example, the provided opinions can help to obtain more accurate conditional independencies.

Acknowledgment

We would like to thank dr. Mechli Imhof-Tas from RadboudUMC, Nijmegen for her valuable help in gathering the opinions from the radiologists regarding the breast cancer network.

References

[1] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine learning*, vol. 9, no. 4, pp. 309–347, 1992.

[2] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Machine learning*, vol. 20, no. 3, pp. 197–243, 1995.

[3] D. M. Chickering, "Optimal structure identification with greedy search," *The Journal of Machine Learning Research*, vol. 3, pp. 507–554, 2003.

[4] L. M. De Campos, "A scoring function for learning Bayesian networks based on mutual information and conditional independence tests," *The Journal of Machine Learning Research*, vol. 7, pp. 2149–2187, 2006.

[5] C. P. De Campos and Q. Ji, "Efficient structure learning of Bayesian networks using constraints," *The Journal of Machine Learning Research*, vol. 12, pp. 663–689, 2011.

[6] S. Huang, J. Li, J. Ye, A. Fleisher, K. Chen, T. Wu, E. Reiman, A. D. N. Initiative *et al.*, "A sparse structure learning algorithm for Gaussian Bayesian network identification from high-dimensional data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 6, pp. 1328–1342, 2013.

[7] M. Studený and D. Haws, "Learning Bayesian network structure: Towards the essential graph by integer linear programming tools," *International Journal of Approximate Reasoning*, vol. 55, no. 4, pp. 1043–1071, 2014.

[8] P. Larrañaga, M. Poza, Y. Yurramendi, R. H. Murga, and C. M. H. Kuijpers, "Structure learning of Bayesian networks by genetic algorithms: A performance analysis of control parameters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 9, pp. 912–926, 1996.

[9] X.-C. Heng, Z. Qin, X.-H. Wang, and L.-P. Shao, "Research on learning Bayesian networks by particle swarm optimization," *Information Technology Journal*, vol. 5, no. 3, pp. 540–545, 2006.

[10] H. Akaike, "A new look at the statistical model identification," *Automatic Control, IEEE Transactions on*, vol. 19, no. 6, pp. 716–723, 1974.

[11] G. Schwarz *et al.*, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[12] P. Spirtes, C. N. Glymour, and R. Scheines, *Causation, prediction, and search*. MIT press, 2000, vol. 81.

[13] R. Robinson, "Counting unlabeled acyclic graphs," in *LNM 622*. Springer, NY, 1977, pp. 220–227.

[14] D. M. Chickering, D. Heckerman, and C. Meek, "Large-sample learning of Bayesian networks is NP-hard," *The Journal of Machine Learning Research*, vol. 5, pp. 1287–1330, 2004.

[15] R. Castelo and A. Siebes, "Priors on network structures. Biasing the search for Bayesian networks," *International Journal of Approximate Reasoning*, vol. 24, no. 1, pp. 39–57, 2000.

[16] M. Richardson and P. Domingos, "Learning with knowledge from multiple experts," in *ICML*, vol. 20, 2003, pp. 624–631.

[17] L. M. de Campos and J. G. Castellano, "Bayesian network learning algorithms using structural restrictions," *International Journal of Approximate Reasoning*, vol. 45, no. 2, pp. 233–254, 2007.

[18] A. Cano, A. R. Masegosa, and S. Moral, "A method for integrating expert knowledge when learning Bayesian networks from data," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 41, no. 5, pp. 1382–1394, 2011.

[19] B. Yet, Z. Perkins, N. Fenton, N. Tai, and W. Marsh, "Not just data: A method for improving prediction with knowledge," *Journal of biomedical informatics*, vol. 48, pp. 28–37, 2014.

[20] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Palo Alto: Morgan Kaufmann Publishers, 1988.

[21] D. M. Chickering, "Learning equivalence classes of Bayesian network structures," *The Journal of Machine Learning Research*, vol. 2, pp. 445–498, 2002.

[22] S. A. Andersson, D. Madigan, and M. D. Perlman, "A characterization of Markov equivalence classes for acyclic digraphs," *Annals of Statistic*, vol. 25, pp. 505–541, 1997.

[23] D. J. MacKay, "Probable networks and plausible predictions-A review of practical Bayesian methods for supervised neural networks," *Network: Computation in Neural Systems*, vol. 6, no. 3, pp. 469–505, 1995.

[24] W. Buntine, "Theory refinement on Bayesian networks," in *Proceedings of the Seventh conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 1991, pp. 52–60.

[25] N. Friedman and D. Koller, "Being Bayesian about network structure. a Bayesian approach to structure discovery in Bayesian networks," *Machine learning*, vol. 50, no. 1-2, pp. 95–125, 2003.

[26] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–38, 1977.

[27] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the EM algorithm," *Applied statistics*, pp. 20–28, 1979.

[28] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *The Journal of Machine Learning Research*, vol. 11, pp. 1297–1322, 2010.

[29] H. Amirkhani, A. Hommersom, M. Rahmati, and P. Lucas, "Improving Bayesian network structure learning using heterogeneous experts," in *International workshop on multi-target prediction*, Nancy, France, September 2014.

[30] H. Amirkhani and M. Rahmati, "Expectation maximization based ordering aggregation for improving the k2 structure learning algorithm," *Intelligent Data Analysis Journal*, vol. 19, no. 2, 2015.

[31] S. L. Lauritzen and D. J. Spiegelhalter, "Local computations with probabilities on graphical structures and their application to expert systems," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 157–224, 1988.

[32] J. Binder, D. Koller, S. Russell, and K. Kanazawa, "Adaptive probabilistic networks with hidden variables," *Machine Learning*, vol. 29, no. 2-3, pp. 213–244, 1997.

[33] I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper, "The Alarm monitoring system: A case study with two probabilistic inference techniques for belief networks," in *Proceedings of the European Conference on Artificial Intelligence in Medicine*, 1989, pp. 247–256.

[34] B. Abramson, J. Brown, W. Edwards, A. Murphy, and R. L. Winkler, "Hailfinder: A Bayesian system for forecasting severe weather," *International Journal of Forecasting*, vol. 12, no. 1, pp. 57–71, 1996.

[35] K. Murphy *et al.*, "The Bayes net toolbox for MATLAB," *Computing science and statistics*, vol. 33, no. 2, pp. 1024–1034, 2001.

[36] P. Leray and O. Francois, "BNT structure learning package: Documentation and experiments," 2004.

[37] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing Bayesian network structure learning algorithm," *Machine learning*, vol. 65, no. 1, pp. 31–78, 2006.

[38] A. R. Masegosa and S. Moral, "An interactive approach for Bayesian network learning using domain/expert knowledge," *International Journal of Approximate Reasoning*, vol. 54, no. 8, pp. 1168–1181, 2013.

[39] M. de Jongh and M. J. Druzdzel, "A comparison of structural distance measures for causal Bayesian network models," *Recent Advances in Intelligent Information Systems, Challenging Problems of Science, Computer Science series*, pp. 443–456, 2009.

[40] M. Velikova, P. Lucas, M. Samulski, and N. Karssemeijer, "A probabilistic framework for image information fusion with an application to mammographic analysis," *Medical Image Analysis*, vol. 16, pp. 865–875, 2012.

[41] ——, "On the interplay of machine learning and background knowledge in image interpretation by Bayesian networks," *Artificial Intelligence in Medicine*, vol. 57, no. 1, pp. 73–86, 2013.

[42] H. Amirkhani and M. Rahmati, "Agreement/disagreement based crowd labeling," *Applied Intelligence*, vol. 41, no. 1, pp. 212–222, 2014.

Hossein Amirkhani received the PhD degree in artificial intelligence from the Computer Engineering Department, Amirkabir University of Technology (Tehran Polytechnic), Iran in 2015 for his work on expert-based structure learning of Bayesian networks. He is currently an assistant professor at the Technology and Engineering Department, University of Qom, Iran. His research interests include machine learning, pattern recognition, and data mining.

Mohammad Rahmati received the MSc in electrical engineering from the University of New Orleans, USA in 1987 and the PhD degree in electrical and computer engineering from University of Kentucky, Lexington, KY USA in 1994. He is currently an associate professor at the Computer Engineering Department, Amirkabir University of Technology (Tehran Polytechnic). His research interests are in the fields of pattern recognition, image processing, bioinformatics, video processing, and data mining. He is the chair of the department and he is also a member of IEEE Signal Processing Society.

Peter J.F. Lucas is a principal investigator with the Institute of Computing and Information Sciences at Radboud University, Nijmegen, and professor at the Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands. His research interests include probabilistic logics, probabilistic graphical models, decision-support systems, and mHealth solutions. Lucas received a PhD in mathematics and computer science from Free University, Amsterdam.

Arjen Hommersom received the PhD degree from the University of Nijmegen for his work on quality assurance of clinical practice guidelines. Currently, he is an assistant professor at the Open University, the Netherlands and a researcher at the Institute of Computing and Information Sciences at Radboud University Nijmegen, the Netherlands. His research interests include knowledge representation and reasoning, in particular probabilistic graphical models and probabilistic logic programming.