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 $2(n^2)$ (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
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 $B = (G, \mathcal{X}, P)$, 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:

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

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 $\theta_i = \prod_{X \in \pi(X_i)} r_j$, 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 $\rho$ be a trail in $G$, i.e., a path without considering the directions of the arcs. A trail $\rho$ is said to be blocked by a set of nodes $Z$ if and only if (at least) one of the following holds:

- $\rho$ contains a chain $U \rightarrow Z_i \rightarrow W$, such that $Z_i$ is in $Z$,
- $\rho$ contains a fork $U \leftarrow Z_i \rightarrow W$, such that $Z_i$ is in $Z$,
- $\rho$ 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 $X \rightarrow Y$ 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
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 $\Theta$ 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., $\theta_{ij} \sim \text{Dir}(\alpha_{ij})$, where $\theta_{ij}$ 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 $\alpha_{ij}$ 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 $\Gamma(x) = \int_{0}^{\infty} e^{-t} t^{x-1} dt$ 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., $P_0(X_i = k, \pi(X_i) = j) = \frac{1}{|\pi(X_i)|}$, 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 $\alpha$, 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 $i$th 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 $(X \leftarrow Y) \in E$,
- $g_i = \leftrightarrow$ 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: $\{\rightarrow, \leftarrow, \leftrightarrow\}$. 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.](image)

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 $g_1 = \rightarrow$, $g_2 = \rightarrow$, $g_3 = \leftarrow$, $g_4 = \leftrightarrow$, $g_5 = \leftrightarrow$, $g_6 = \leftrightarrow$.

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

The experts’ opinions are indicated by $R$. The opinion of the $i$th expert regarding the $j$th pair is denoted by $O_{ij} \in \{0, \rightarrow, \leftarrow, \leftrightarrow\}$, where $O_{ij} = \emptyset$ meaning that the $i$th expert has not provided any opinion about the $j$th pair. We use $O^i$ to indicate all opinions provided by the $i$th expert, $O_j$ to mention the opinions provided by all experts about the $j$th pair, and $O$ to denote all provided opinions.

We model the accuracy of an expert by three parameters:

- $\gamma_1$: The probability of detecting the existing edges with correct directions,
- $\gamma_2$: The probability of detecting the existing edges with reverse directions,
- $\gamma_3$: The probability of correctly detecting the absent edges.

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

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

$$
\begin{align*}
\rightarrow & \quad \leftarrow & \quad \leftrightarrow \\
0.6 & \quad 0.1 & \quad 0.3 \\
0.1 & \quad 0.6 & \quad 0.3 \\
0.1 & \quad 0.1 & \quad 0.8
\end{align*}
$$

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 $g_{ij} = \leftrightarrow$, 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 $D$,
and experts’ opinions $O$. In this model, $K$ denotes the set of information determining the structure. Note that the prior distribution $p$ 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 $\epsilon$. 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, $O_E^i$ and $O_A^i$ are the opinions of the $i$th expert about the existing and absent edges in $G$, respectively. Based on this model, the opinions regarding existing edges are influenced by $\gamma_1$ and $\gamma_2$ parameters, and the opinions regarding absent edges are influenced by the $\gamma_3$ 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 $\gamma$ are independent given $G$. Note that based on Fig. 2, if $O$ is not given, $D$ and $\gamma$ are independent, regardless of whether $G$ is given or not. Anyway, having $G$ does not violate this independence, and because we need $D \perp \gamma | G$ in the subsequent sections, we introduce this statement instead of $D \perp \gamma$. This also holds for some other statements in Table 1.

### 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.

---

**Table 1**

<table>
<thead>
<tr>
<th>Number</th>
<th>Statement</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$G \perp \gamma$</td>
<td>Fig. 2</td>
</tr>
<tr>
<td>2</td>
<td>$G \perp \gamma</td>
<td>p$</td>
</tr>
<tr>
<td>3</td>
<td>$D \perp G</td>
<td>G$</td>
</tr>
<tr>
<td>4</td>
<td>$O \perp D</td>
<td>G$</td>
</tr>
<tr>
<td>5</td>
<td>$O \perp D</td>
<td>G, \gamma$</td>
</tr>
<tr>
<td>6</td>
<td>$O^i \perp O^j</td>
<td>G$</td>
</tr>
<tr>
<td>7</td>
<td>$O^i \perp O^j</td>
<td>G, \gamma$</td>
</tr>
<tr>
<td>8</td>
<td>$O^i \perp \gamma^3</td>
<td>G, \gamma^3$</td>
</tr>
<tr>
<td>9</td>
<td>$O^i \perp O^j</td>
<td>G, \gamma^3$</td>
</tr>
<tr>
<td>10</td>
<td>$O^i \perp g_x</td>
<td>y^j, \gamma^3$</td>
</tr>
<tr>
<td>11</td>
<td>$O^i \perp O^j</td>
<td>g_x, p, \gamma^3$</td>
</tr>
<tr>
<td>12</td>
<td>$O^i \perp \gamma^i</td>
<td>g_x, p, \gamma^3$</td>
</tr>
<tr>
<td>13</td>
<td>$O^i \perp p</td>
<td>g_x, \gamma^3$</td>
</tr>
<tr>
<td>14</td>
<td>$(\gamma_1^i, \gamma_2^i) \perp \gamma_3^i$</td>
<td>Fig. 4</td>
</tr>
</tbody>
</table>
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 \(\gamma\). A Bayesian measure of the goodness of fit of \(G\) is its posterior probability given \(D, O, \gamma\):

\[
P(G \mid D, O, \gamma) \propto P(D \mid G, \gamma) P(G \mid \gamma) P(O \mid G, D, \gamma).
\]

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

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

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 the decision tree depicted in Fig. 5. Note that we only use 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 \((p, \gamma)\). 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

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

Since \(g_i\) is a member of \(\{\rightarrow, \leftrightarrow, \leftarrow\}\), we can write

\[
\log P(O_i, g_i \mid p, \gamma) = \sum_{k \in \{\rightarrow, \leftrightarrow, \leftarrow\}} I(g_i = k) \log P(O_i, g_i = k \mid p, \gamma),
\]

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

\[
P(O_i, g_i = k \mid p, \gamma) = P(g_i = k \mid p, \gamma) P(O_i \mid g_i = k, p, \gamma).
\]

Using statement 2 in Table 1, we have

\[
P(g_i = k \mid p, \gamma) = P(g_i = k \mid p) = p_k. \quad (3)
\]

Also according to statement 11 in Table 1, we have

\[
P(O_i \mid g_i = k, p, \gamma) = \prod_{j=1}^{R} P(O^j_i \mid g_i = k, p, \gamma). \quad (4)
\]

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

\[
P(O^j_i \mid g_i = k, p, \gamma) = P(O^j_i \mid g_i = k, \gamma^j),
\]

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

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

\[
\log P(O, G \mid p, \gamma) = \sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftrightarrow, \leftarrow\}} I(g_i = k) \log p_k + \sum_{j=1}^{R} \log P(O^j_i \mid g_i = k, \gamma^j). \quad (5)
\]

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.

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.

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 \(p\) and the experts’ accuracies \(\gamma\) as the parameters, the maximum-likelihood estimate of these parameters is

\[
(p, \gamma) = \arg\max_{p, \gamma} \{\log P(O \mid p, \gamma)\}.
\]
The conditional expectation of the complete log-likelihood (5) given opinions $O$ and the current estimate of the parameters $p^{(t)}, \gamma^{(t)}$ is

$$E_\eta[\log P(O, G \mid p, \gamma)] = \sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow, \leftrightarrow\}} \left\{ E_\eta[I(g_i = k)] \times \left( \log p_k + \sum_{j=1}^{R} \log P(O_j^{(t)} \mid g_i = k, \gamma) \right) \right\}, \quad (6)$$

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

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

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

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

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

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

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

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:

$$p_{k}^{(t+1)} = \frac{1}{N} \sum_{i=1}^{N} E_\eta[I(g_i = k)],$$

$$(\gamma_1^{(t+1)}) = \frac{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow, \leftrightarrow\}} E_\eta[I(g_i = k)] \times I(O_i^{(t)} = k)}{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} E_\eta[I(g_i = k)] \times I(O_i^{(t)} \neq \emptyset)},$$

$$(\gamma_2^{(t+1)}) = \frac{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} E_\eta[I(g_i = k)] \times I(O_i^{(t)} = \emptyset)}{\sum_{i=1}^{N} \sum_{k \in \{\rightarrow, \leftarrow\}} E_\eta[I(g_i = k)] \times I(O_i^{(t)} \neq \emptyset)}, \quad (8)$$

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

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

(i) Take initial estimates of the parameters $p, \gamma$.

(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...
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:

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

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,

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

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

$$P(O^i \mid G) = \int_{0}^{1} \int_{0}^{1-\gamma_1} \int_{0}^{1-\gamma_2} \left( P(\gamma_1, \gamma_2, \gamma_3 \mid G) \times P(O^i \mid \gamma_1, \gamma_2, \gamma_3, G) \right) d\gamma_3 d\gamma_2 d\gamma_1. \quad (10)$$

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

According to statements 1 and 14 in Table 1,

$$P(\gamma_1, \gamma_2, \gamma_3 \mid G) = P(\gamma_1, \gamma_2)P(\gamma_3). \quad (11)$$

In addition, based on statement 9 we have

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

which can be written as

$$P(O^i \mid \gamma_1, \gamma_2, \gamma_3, G) = (\gamma_1)^{n_1}(\gamma_2)^{n_2}(1-\gamma_1-\gamma_2)^{n_3}(\gamma_3)^{m_1} \left( \frac{1-\gamma_3}{2} \right)^{m_2}, \quad (12)$$

where

- $n_1$ is the number of existing edges in $G$ that are detected by expert $i$ with correct directions,
- $n_2$ is the number of existing edges in $G$ that are detected by expert $i$ with reverse directions,
- $n_3$ is the number of existing edges in $G$ that are mentioned as absent edges by expert $i$,
- $m_1$ is the number of absent edges in $G$ that are correctly detected by expert $i$,
- $m_2$ 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

$$P(O^i \mid G) = \left( \int_{0}^{1} P(\gamma_3) (\gamma_3)^{m_1} \left( \frac{1-\gamma_3}{2} \right)^{m_2} d\gamma_3 \right) \times \left( \int_{0}^{1} \int_{0}^{1-\gamma_1} \int_{0}^{1-\gamma_2} P(\gamma_1, \gamma_2) (\gamma_1)^{n_1}(\gamma_2)^{n_2}(1-\gamma_1-\gamma_2)^{n_3} d\gamma_1 d\gamma_2 d\gamma_3 \right). \quad (13)$$

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

$$I_1^i = \int_{0}^{1} P(\gamma_3) (\gamma_3)^{m_1} \left( \frac{1-\gamma_3}{2} \right)^{m_2} d\gamma_3,$n_3-1\right) d\gamma_1 d\gamma_2 d\gamma_3. \quad (14)$$

An appropriate distribution for $P(\gamma_3)$ in $I_1^i$ is the Beta distribution. If the shape parameters of this distribution are denoted by $\beta_{n_1}$ and $\beta_{n_2}$, we have

$$P(\gamma_3) = \frac{1}{B(\beta_{n_1}, \beta_{n_2})} (\gamma_3)^{\beta_{n_1}-1}(1-\gamma_3)^{\beta_{n_2}-1}, \quad (15)$$

where

$$B(\beta_{n_1}, \beta_{n_2}) = \int_{0}^{1} t^{\beta_{n_1}-1}(1-t)^{\beta_{n_2}-1} dt \quad (16)$$

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

$$I_1^i = \frac{\beta_{n_1}+m_1-1}{2m_2} \times \frac{B(\beta_{n_1}+m_1-1, \beta_{n_2})}{B(\beta_{n_1}, \beta_{n_2})}, \quad (17)$$

which can be written as

$$I_1^i = \frac{B(\beta_{n_1}+1, \beta_{n_2}+1)}{2m_2} \times \frac{B(\beta_{n_1}+m_1-1, \beta_{n_2})}{B(\beta_{n_1}, \beta_{n_2})} \times \frac{1}{(\beta_{n_1}+m_1-1)} \times \frac{1}{(\beta_{n_2}+m_2-1)}.$$
and therefore,

\[
I^2_2 = \frac{1}{\text{B}(\alpha_1, \alpha_2, \alpha_3)} \times \text{B}(\alpha_1, \alpha_2, \alpha_3) \times \text{B}(\alpha_1 + \alpha_2, \alpha_3 + \alpha_3) \times \text{B}(\alpha_1 + \alpha_3, \alpha_2 + \alpha_3 + \alpha_3, \alpha_3).
\]

which can be written as

\[
I^2_2 = \frac{1}{\text{B}(\alpha_1, \alpha_2, \alpha_3)} \times \text{B}(\alpha_1, \alpha_2, \alpha_3) \times \text{B}(\alpha_1 + \alpha_2, \alpha_3 + \alpha_3) \times \text{B}(\alpha_1 + \alpha_3, \alpha_2 + \alpha_3 + \alpha_3, \alpha_3). \tag{17}
\]

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

\[
\log P(O^i | G) = \log \text{B}(\beta_1 + m_1, \beta_2 + m_2) - m_1 \log 2 - \log \text{B}(\beta_1, \beta_2) + \log \text{B}(\alpha_1 + n_1, \alpha_2 + n_2 + \alpha_3 + n_3) + \log \text{B}(\alpha_1 + n_1, \alpha_2 + n_2 + \alpha_3 + n_3) - \log \text{B}(\alpha_1, \alpha_2, \alpha_3).
\]

Finally, based on equation (9) we get \(P(O^i | G)\) 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].

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 \(\gamma\) and \(\gamma_3\) means more accurate experts, these parameters have higher values in the “good” population. On the other hand, since higher \(\gamma_2\) means less accurate experts, this parameter has higher values in the “weak” population.

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.

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.

### Table 2

**Description of the Networks Used in the Simulation Experiments**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Nodes</th>
<th>Edges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia</td>
<td>Diagnosing some respiratory diseases</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>Insurance</td>
<td>Evaluating car insurance risks</td>
<td>27</td>
<td>52</td>
</tr>
<tr>
<td>Alarm</td>
<td>Monitoring patients in intensive care</td>
<td>37</td>
<td>46</td>
</tr>
<tr>
<td>Hailfinder</td>
<td>Predicting summer hailstorms in northern Colorado</td>
<td>56</td>
<td>66</td>
</tr>
</tbody>
</table>

### Table 3

**The Accuracy Parameters Assigned to Experts in Simulated Populations**

<table>
<thead>
<tr>
<th>Populations</th>
<th>Weak</th>
<th>Mediocre</th>
<th>Good</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\gamma_1)</td>
<td>0.15</td>
<td>0.80</td>
<td>0.85</td>
</tr>
<tr>
<td>(\gamma_2)</td>
<td>0.30</td>
<td>0.30</td>
<td>0.70</td>
</tr>
<tr>
<td>(\gamma_3)</td>
<td>0.75</td>
<td>0.10</td>
<td>0.90</td>
</tr>
<tr>
<td>(\gamma_{10})</td>
<td>0.40</td>
<td>0.25</td>
<td>0.50</td>
</tr>
<tr>
<td>(\gamma_{11})</td>
<td>0.45</td>
<td>0.35</td>
<td>0.45</td>
</tr>
<tr>
<td>(\gamma_{12})</td>
<td>0.55</td>
<td>0.20</td>
<td>0.60</td>
</tr>
<tr>
<td>(\gamma_{13})</td>
<td>0.60</td>
<td>0.20</td>
<td>0.80</td>
</tr>
<tr>
<td>(\gamma_{14})</td>
<td>0.70</td>
<td>0.15</td>
<td>0.95</td>
</tr>
<tr>
<td>(\gamma_{15})</td>
<td>0.80</td>
<td>0.05</td>
<td>0.80</td>
</tr>
<tr>
<td>(\gamma_{16})</td>
<td>0.90</td>
<td>0.20</td>
<td>0.70</td>
</tr>
<tr>
<td>(\gamma_{17})</td>
<td>0.70</td>
<td>0.30</td>
<td>0.65</td>
</tr>
<tr>
<td>(\gamma_{18})</td>
<td>0.80</td>
<td>0.10</td>
<td>0.90</td>
</tr>
</tbody>
</table>

Mean | 0.59 | 0.33 | 0.51 |
| 0.30 | 0.40 | 0.30 |
| 0.60 | 0.30 | 0.65 |
| 0.80 | 0.25 | 0.67 |
| 0.61 | 0.24 | 0.78 |

0162-8828 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
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 $\gamma_1$ and $\gamma_2$ parameters of all experts are set to one, and $\gamma_2$ 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 in the simulation experiments. More precisely, if the opinions are generated from the accuracy parameters, it uses the mean of true accuracies for the used Bayesian networks are sparse. The real world Bayesian network structures are sparse. The third worst results for the weak populations, the second worst results (after the Expert scenario) reduces the accuracy of the obtained results.

In all of the above scoring functions, for the log-likelihood $\log P(D | G)$, 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 $\nu$ 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, $\nu$}, 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

---

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

<table>
<thead>
<tr>
<th>BN</th>
<th>$p_{+-}$</th>
<th>$p_{-+}$</th>
<th>$p_{--}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia</td>
<td>0.11</td>
<td>0.18</td>
<td>0.71</td>
</tr>
<tr>
<td>Insurance</td>
<td>0.07</td>
<td>0.08</td>
<td>0.85</td>
</tr>
<tr>
<td>Alarm</td>
<td>0.04</td>
<td>0.03</td>
<td>0.93</td>
</tr>
<tr>
<td>Hailfinder</td>
<td>0.02</td>
<td>0.02</td>
<td>0.96</td>
</tr>
</tbody>
</table>
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 the 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 Hallfinder network in the Table 8.

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...
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 $c$.

### 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. Two 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.
In order to reduce the effect of randomness in the reported data which is selected from the set section 6.1.1. In this table, features learned using the scoring functions mentioned in sub-
distances between the gold-standard structure and the struc-
structure.
shown in Fig. 6, which is considered as the gold-standard analysis procedure.
In addition, the Bayesian network integrates the results of microcalcification analysis, which is a separate image
clinical features with radiological examination by X-rays.
is shown in Fig. 6. The Bayesian network model combines
works we had designed previously. One of those networks
radiologists we asked for our current experiment. None
enced radiologists. These radiologists were different from
Bayesian network structures in collaboration with experi-
TABLE 11
The Accuracy Parameters and the Volume of Opinions Provided by Different Experts in the Breast Cancer Experiment

<table>
<thead>
<tr>
<th></th>
<th>Accuracy Parameters</th>
<th>Volume of Opinions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\gamma_1)</td>
<td>(\gamma_2)</td>
</tr>
<tr>
<td>1</td>
<td>0.17</td>
<td>0.08</td>
</tr>
<tr>
<td>2</td>
<td>0.44</td>
<td>0.11</td>
</tr>
<tr>
<td>3</td>
<td>0.50</td>
<td>0.08</td>
</tr>
<tr>
<td>4</td>
<td>0.50</td>
<td>0.17</td>
</tr>
<tr>
<td>5</td>
<td>0.59</td>
<td>0.24</td>
</tr>
<tr>
<td>6</td>
<td>0.75</td>
<td>0.17</td>
</tr>
<tr>
<td>7</td>
<td>0.91</td>
<td>0.09</td>
</tr>
<tr>
<td><strong>Avg</strong></td>
<td>0.35</td>
<td>0.13</td>
</tr>
</tbody>
</table>

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 \(\beta_1, \beta_2, \alpha_{11}, \alpha_{12}, \alpha_{13}\) 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.

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


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.