A Score-and-Search Approach to Learning Bayesian Networks with Noisy-OR Relations

Charupriya Sharma C9SHARMA@UWATERLOO.CA

Zhenyu A. Liao Z6LIAO@UWATERLOO.CA
David R. Cheriton School of Computer Science, University of Waterloo, Canada

James Cussens JAMES.CUSSENS@BRISTOL.AC.UK
Department of Computer Science, University of Bristol, UK

Peter van Beek VANBEEK@UWATERLOO.CA
David R. Cheriton School of Computer Science, University of Waterloo, Canada

Abstract

A Bayesian network is a probabilistic graphical model that consists of a directed acyclic graph (DAG), where each node is a random variable and attached to each node is a conditional probability distribution (CPD). A Bayesian network can be learned from data using the well-known score-and-search approach, and within this approach a key consideration is how to simultaneously learn the global structure in the form of the underlying DAG and the local structure in the CPDs. Several useful forms of local structure have been identified in the literature but thus far the score-and-search approach has only been extended to handle local structure in form of context-specific independence. In this paper, we show how to extend the score-and-search approach to the important and widely useful case of noisy-OR relations. We provide an effective gradient descent algorithm to score a candidate noisy-OR using the widely used BIC score and we provide pruning rules that allow the search to successfully scale to medium sized networks. Our empirical results provide evidence for the success of our approach to learning Bayesian networks that incorporate noisy-OR relations.

Keywords: Bayesian networks; structure learning; causal noisy-OR.

1. Introduction

Bayesian networks (BNs) are widely used probabilistic graphical models with applications in knowledge discovery, decision support, and prediction (Darwiche, 2009; Koller and Friedman, 2009). A BN can be learned from data using the well-known *score-and-search* approach, where a scoring function is used to evaluate the fit of a proposed BN to the data in the space of directed acyclic graphs (DAGs). Current implementations of this approach such as (Yuan et al., 2011), (Bartlett and Cussens, 2013), and (van Beek and Hoffmann, 2015) consider only conditional probability tables (CPTs) as representations for the underlying conditional probability distributions (CPDs) for discrete variables. However, the size of the CPT for a variable grows exponentially as the number of parents increases. For example, the CPT of a binary child node with n binary parents requires 2^{n+1} probabilities. This presents a practical difficulty in parameter estimation and inference and has motivated many structured representations for CPDs that exploit the relationship between a child and its parents and aim at reducing model complexity.

A widely used local structure is the noisy-OR relation (Good, 1961; Pearl, 1988) and its generalizations such as leaky noisy-OR (Henrion, 1987) and noisy-MAX (Díez, 1993). These relations model the CPD over causes (parents) and effects (children). The noisy-OR assumes a form of causal

independence (CI) and allows one to specify a CPT with just n parameters instead of 2^{n+1}. Zhang and Poole (1996) derived variable elimination under CI and demonstrated the advantage of CI in inference. Besides CI, Boutilier et al. (1996) proposed a decision tree model that captures context-specific independence (CSI). Later, Chickering et al. (1997) extended the tree structure to decision graphs that encode equality constraints and Poole and Zhang (2003) derived a version of variable elimination under CSI. Despite showing advantages in inference, these studies—with the exception of Chickering et al. (1997)—only consider the local structure of CPDs while assuming some fixed global structure; i.e., the underlying DAG for the BN is fixed and some or all of the CPTs are replaced with locally structured representations.

However, when some or all of the CPTs within some fixed global structure are replaced by locally structured representations with reduced complexity, the existing DAG structure is often not optimal or appropriate for the new representations anymore. Consider the Bayesian information criterion (BIC) that consists of the log likelihood of the data being generated by the model and a penalty for model complexity. The structured CPDs are likely to reduce the likelihood due to the so-called compression error (Xiang and Baird, 2018; Zagorecki and Druzdzel, 2013), but they also have a smaller penalty as a result of using fewer parameters. These changes open up the opportunity for some alternative global structures to have better scores. Ideally, the learning algorithm should be able to choose, for example, between a CPD represented as a CPT with a smaller number of parents, and a CPD approximated as a noisy-OR with a larger number of parents.

Assuming a fixed global structure may also lead to inaccuracies when assessing the effect of using structured representations. Compression error only measures the ability of a new representation to reproduce the original CPT, but that is not the goal of BNs. For example, the CPD modeled by noisy-OR may be different from the CPD by CPT, but with a different structure the former might be a better fit for the distribution of the data. Similarly, measuring inference error with a fixed structure is misleading. Failing to consider new structures to better accommodate alternative representations makes the false impression that we trade some posterior accuracy for reduced complexity, although in practice the posterior accuracy may even be improved with proper structure learning.

Friedman and Goldszmidt (1998) are the first to incorporate local structures in Bayesian network structure learning (BNSL) with the *score-and-search* approach. They show that using structured representations in hill-climbing allows the learning algorithm to explore more complex networks and thus avoids inferring incorrect conditional independence relations. The observation is also supported by Talvitie et al. (2019) in an exact search to find the optimal BN using a tree structure. Their experiments, albeit with some explicit structural constraints on the underlying DAG, suggest that structured CPDs can help the search algorithm find correct BNs with fewer samples, especially on real-world datasets. However, they also find that for some datasets CPT can still perform better. The discrepancy is likely attributed to the fact that all proposed structured representations are used separately as the sole representation for the CPDs. If the structured representations are compared with CPTs and are only used when appropriate, they can then better help the search algorithm to find the correct structure and maintain the complexity advantage in inference.

In this paper, we propose the first score-and-search approach for learning Bayesian networks with both CPT and noisy-OR relations as possible representations for CPDs. Importantly, we simultaneously learn both the global structure in the form of the underlying DAG and the local structure in the CPDs, we place no a priori constraints on the global structure, and we exactly determine all networks within a given factor of optimal. Our approach has two primary advantages. First, our approach only replaces a CPT with a noisy-OR relation when it is appropriate. Converting an ar-

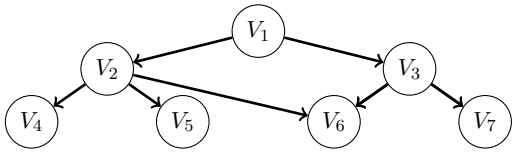

Figure 1: Example Bayesian network: Each variable has the state space $\{0, 1\}$. Consider the parent set of V_6, $\Pi_6 = \{V_2, V_3\}$ The state space of Π_6 is $\Omega_{\Pi_6} = \{\{0, 0\}, \{0, 1\}, \{1, 0\}, \{1, 1\}\}$. and $r_{\Pi_6} = 4$.

bitrary proportion of CPTs to structured representations can lead to significant degradation of the expressive power of the model, and it is difficult to determine the optimal proportion a priori. Our approach controls the degradation by specifying a Bayes factor (BF) (Kass and Raftery, 1995) that measures how far a BN can deviate from the optimal network, and so only near-optimal networks with both CPTs and noisy-OR relations are learned in a principled manner. Second, our approach can scale to BNs of moderate sizes. Even local structure modelling with structured representations such as (Xiang, 2019) suffers from a large search space. Our approach, on the other hand, can effectively prune most candidate parent sets of a variable by leveraging the results from learning BNs with CPTs given a BF (Liao et al., 2019). We empirically demonstrate that our approach can learn these *mixed* BNs in a principled manner that takes advantage of a reduced complexity.

2. Background

In this section, we review Bayesian networks (Koller and Friedman, 2009; Darwiche, 2009), noisy-OR relations (Good, 1961; Pearl, 1988) and the BIC scoring function (Lam and Bacchus, 1994; Schwarz, 1978).

2.1 Bayesian Networks

A Bayesian network (BN) is a probabilistic graphical model that consists of a labeled directed acyclic graph (DAG), $\mathbf{G} = (\mathbb{V}, \mathbb{E})$ in which the nodes $\mathbb{V} = \{V_1, \ldots, V_n\}$ correspond to random variables, the edges $\mathbb{E}$ represent direct influence of one random variable on another, and each node V_i is labeled with a conditional probability distribution $P(V_i \mid \Pi_i)$ that specifies the dependence of the variable V_i on its set of parents Π_i in the DAG. A BN can alternatively be viewed as a factorized representation of the joint probability distribution over the random variables and as an encoding of the Markov condition on the nodes; i.e., given its parents, every variable is conditionally independent of its non-descendants.

In this paper, we assume that each random variable V_i is binary. Each Π_i has state space of a set of candidate instantiations of the nodes in Π_i, $\Omega_{\Pi_i} = \{\pi_{i1}, \ldots, \pi_{ir_{\Pi_i}}\}$. We use $r_{\Pi_i} = 2^{|\Pi_i|}$ to refer to the number of possible instantiations of the parent set Π_i of V_i (see Figure 1). The set $\theta = \{\theta_{ijk}\}$ for all $i = \{1, \ldots, n\}, j = \{1, \ldots, r_{\Pi_i}\}$ and $k = \{0, 1\}$ represents parameter estimates in G obtained from a dataset, where each θ_{ijk} estimates the conditional probability $P(V_i = k \mid \Pi_i = \pi_{ij})$. Given a node V_i and a parent set Π_i, we define the set $\theta_i := \big\{\theta_{ijk} \mid j \in \{1, \ldots, r_{\Pi_i}\}, k \in \{0, 1\}\big\}$. We refer to θ_i as the *full CPT* of node i.

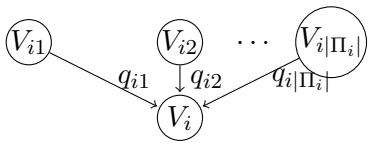

Figure 2: Causal structure for a Bayesian network with a noisy-OR relation, where the set of causes $\Pi_i := \{V_{i1}, \ldots, V_{i|\Pi_i|}\}$ leads to effect V_i and there is a noisy-OR relation at node V_i.

The predominant method for Bayesian network structure learning (BNSL) from data is the *score-and-search* method. Let $\mathbf{G}$ be a DAG over random variables $\mathbb{V}$, and let $I = \{I_1, \ldots, I_N\}$ be a dataset, where each instance I_i is an n-tuple that is a complete instantiation of the variables in $\mathbb{V}$. A *scoring function* $\sigma(\mathbf{G} \mid I)$ assigns a real value measuring the quality of $\mathbf{G}$ given the data I. Without loss of generality, we assume that a lower score represents a better quality network structure. To simplify notation, we use $\sigma(\mathbf{G})$ in place of $\sigma(\mathbf{G} \mid I)$ when the data is clear from context. In this paper, we focus on solving the problem of ϵ-Bayesian Network Structure Learning (ϵBNSL) (Liao et al., 2019).

Definition 1 *Given a non-negative constant ϵ, a dataset $I = \{I_1, \ldots, I_N\}$ over random variables $\mathbb{V} = \{V_1, \ldots, V_n\}$ and a scoring function σ, the ϵ-Bayesian Network Structure Learning (ϵBNSL) problem is to find all* ***credible networks****, which are all networks that have a score $\sigma(\mathbf{G})$ such that $OPT \leq \sigma(\mathbf{G} \mid I) \leq OPT + \epsilon$, where OPT is the score of the optimal Bayesian network.*

It has been shown in (Liao et al., 2019) that a good choice for ϵ is $\log BF$. By specifying the constant ϵ in terms of a Bayes factor, we can control the level of tolerance for network degradation and learn all near-optimal networks with both CPTs and noisy-OR relations as best determined by the trade-off between the fit with the data and the complexity of the model.

2.2 BIC/MDL Scoring Function

In this work, we focus on the Bayesian information criterion (BIC) scoring function. As the BIC function is decomposable, when the θ_i is given we can associate a score to a candidate parent set Π_i of V_i as follows,

$$BIC : \sigma(\Pi_i) = -L(\theta_i) + t(\Pi_i) \cdot w, \tag{1}$$

where the formula consists of a term measuring the likelihood of the candidate parent set given the data and a penalty term for the number of parameters needed to specify the full CPT for the candidate parent set. Here, $L(\theta_i) = \sum_{j=1}^{r_{\Pi_i}} \sum_{k \in \{0,1\}} n_{ijk} \log \theta_{ijk}$, n_{ijk} is the number of instances in dataset I where $V_i = k$ and $\Pi = \pi_{ij}$ co-occur, and $t(\Pi_i) = 2^{|\Pi_i|}$. The penalty term is weighted by $w = \log(N)/2$ where N is the number of instances in dataset I. Note that $\sigma(\mathbf{G}) = \sum_{i=1}^{n} \sigma(\Pi_i)$. We use the natural logarithm throughout the paper.

2.3 Patterns for CPTs: Noisy-OR

With the noisy-OR relation one assumes that there are a set of causes $\Pi_i := \{V_{i1}, \ldots, V_{i|\Pi_i|}\}$ leading to an effect V_i, where $V_i, V_{ij} \in \mathbb{V}$ for all $j \in \{1, ...|\Pi_i|\}$ and $V_i \notin \Pi_i$ (see Figure 2). Each cause $V_{ij} \in \Pi_i$ is either present or absent, and each V_{ij} in isolation is likely to cause V_i and the

likelihood is not diminished if more than one cause is present. Further, one assumes that all possible causes are given and when all causes are absent, the effect is absent. Finally, one assumes that the mechanism or reason that inhibits a V_{ij} from causing V_i is independent of the mechanism or reason that inhibits a $V_{ij'}$, $j' \neq j$, from causing V_i.

For a node V_i and parent set Π_i, a noisy-OR relation specifies a CPT using $|\Pi_i|$ parameters, $\mathbf{q}_i = q_{i1}, \ldots, q_{i|\Pi_i|}$, one for each parent, where q_{ij} is the probability that V_i is false given that V_{ij} is true and all of the other parents are false,

$$P(V_i = 0 \mid V_{ij} = 1, V_{ij'} = 0_{[\forall j', j' \neq j]}) = q_{ij}.$$

From these parameters, the full CPT representation of size 2^{n+1} can be generated using,

$$\phi_{ij0} = \begin{cases} \prod_{j \in T_x} q_{ij} & \text{if } T_x \neq \{\} \\ 1 & \text{otherwise,} \end{cases} \tag{2}$$

where $T_x = \{j \mid V_{ij} = 1\}$. The last condition (when T_x is empty) corresponds to the assumptions that all possible causes are given and that when all causes are absent, the effect is absent; i.e., $P(V_i = 0 \mid V_{i1} = 0, \ldots, V_{i|\Pi_i|} = 0) = 1$. Of course, $\phi_{ij1} = 1 - \phi_{ij0}$. The set $\phi_i := \{\phi_{ijk} \mid j \in \{1, \ldots, r_{\Pi_i}\}, k \in \{0, 1\}\}$ is referred to as the *noisy-OR CPT* of node i.

The above assumptions are not as restrictive as may first appear. One can always introduce an additional random variable V_{i0} that is a parent of V_i but itself has no parents. The variable V_{i0} represents all of the other reasons that could cause V_i to occur. The node V_{i0} and the prior probability $P(V_{i0})$ are referred to as a *leak node* and the *leak probability*, respectively. In this work we assume that all the causes are known.

3. Our Solution

In this section, we present our *score-and-search* approach for learning all Bayesian networks, given local scores, that are within a given factor ϵ of optimal, where the networks can contain both full CPT and noisy-OR relations as possible representations for the CPDs. In general, a score-and-search approach scores candidate parent sets for the nodes in the network and searches for the choice of a parent set, one for each node, that leads to the best overall score while ensuring that the network is acyclic. Before presenting our overall approach for solving ϵBNSL (Section 3.3), we first describe an effective gradient descent algorithm to score a candidate noisy-OR relation using the widely used BIC score (Section 3.1) and pruning rules that allow the search to scale to larger networks (Section 3.2).

3.1 BIC Score for Noisy-OR Relations

The BIC score consists of a maximum likelihood term and a penalty term. We present a gradient descent algorithm that is based on minimizing a KL divergence as it is known that minimizing the KL divergence results in maximizing the likelihood (see, e.g., Murphy (2012)). Recall that the elements of θ_i are conditional probabilities computed from the dataset I. Given a node V_i, we must compute maximum likelihood estimates for the noisy-OR CPT ϕ_i for every candidate parent set Π_i, such that the conditional KL divergence between the full CPT θ_i and the resulting noisy-OR CPT ϕ_i that is determined by the $\mathbf{q}_i$ (see Equation 2), is minimized. Note that the KL divergence between

two *conditional* probability distributions, $P(A|B)$ and $Q(A|B)$ is given by,

$$D_{KL}(P(A|B) \mid\mid Q(A|B)) = \sum_{b \in B} P(B=b) \sum_{a \in A} P(A=a|B=b) \log \frac{P(A=a|B=b)}{Q(A=a|B=b)}.$$

We note that an alternative approach to estimate noisy-OR parameters is to maximize the log-likelihood using the expectation-maximization (EM) technique, which was derived in Dempster et al. (1977) and applied to noisy-OR in Vomlel (2006). We perform an experimental comparison of the two approaches in Section 4.

To derive our gradient descent algorithm, we begin with the definition of KL divergence for the two conditional probability distributions, θ_i and ϕ_i, and rewrite it into a more convenient form:

$$\begin{aligned}
D_{KL}(\theta_i \mid\mid \phi_i) &\overset{0}{=} \sum_{j=1}^{r_{\Pi_i}} P(\pi_{ij}) \sum_{k \in \{0,1\}} \theta_{ijk} \log \frac{\theta_{ijk}}{\phi_{ijk}} \\
&\overset{1}{=} \sum_{j=1}^{r_{\Pi_i}} \frac{n_{ij}}{N} \sum_{k \in \{0,1\}} \theta_{ijk} \log \frac{\theta_{ijk}}{\phi_{ijk}} \\
&\overset{2}{=} \frac{1}{N} \sum_{j=1}^{r_{\Pi_i}} \sum_{k \in \{0,1\}} n_{ijk} \cdot \theta_{ijk} \log \frac{\theta_{ijk}}{\phi_{ijk}} \\
&\overset{3}{=} \frac{1}{N} \sum_{j=1}^{r_{\Pi_i}} \sum_{k \in \{0,1\}} n_{ijk} \cdot \theta_{ijk} \log \theta_{ijk} - \frac{1}{N} \sum_{j=1}^{r_{\Pi_i}} \sum_{k \in \{0,1\}} n_{ijk} \cdot \theta_{ijk} \log \phi_{ijk},
\end{aligned}$$

where N is the number of instances in our dataset. To find ϕ_i such that $D_{KL}(\theta_i \mid\mid \phi_i)$ is minimized, note that the first term in Step 3 is constant. So, we must determine,

$$\underset{\mathbf{q_i}}{\operatorname{argmin}}\, D_{KL}(\theta_i \mid\mid \phi_i) = -\sum_{j=1}^{r_{\Pi_i}} \sum_{k \in \{0,1\}} n_{ijk} \cdot \theta_{ijk} \log \phi_{ijk},$$

where the $\mathbf{q}_i$ that minimizes the KL divergence are the maximum likelihood estimates for ϕ_i that are determined by the $\mathbf{q}_i$ (Equation 2). The penalty term in the BIC score can be computed in constant time; specifically, the number of parents in the candidate parent set. Thus, fitting these noisy-OR parameters gives us the BIC score for the noisy-OR for a candidate parent set. To find these noisy-OR parameters, we use Algorithm 1 which performs gradient descent for the derivative,

$$\Delta_{KL}^{\mathbf{q}_i} = \frac{d}{d\mathbf{q}_i} \sum_{j=1}^{r_{\Pi_i}} \sum_{k \in \{0,1\}} n_{ijk} \cdot \log \phi_{ijk}. \tag{3}$$

We start with an initial guess for the set of noisy-OR parameters $\mathbf{q}_i$ and evaluate term $\Delta_{KL}^{\mathbf{q}_i}$ for these values (Equation 3). The initial guess uses hot starts in that the solution for a smaller candidate parent set is used as the starting point when estimating the parameters for a candidate set that is a superset. We perform gradient descent over $\mathbf{q}_i$, where each step update is found by a simple geometric line search algorithm (see Algorithm 1). Geometric line search is a backtracking line search procedure, where we first choose a descent direction and then determine the maximum amount to move along that direction.

Algorithm 1 Computing Noisy-OR Parameters for a Candidate Parent Set

Input: Node V_i, candidate set Π_i, a dataset I of N instances.
Parameter: Threshold t, maximum iterations $maxIter$
Output: A set of noisy-OR parameters : $\mathbf{q}_i = q_{i1}, \ldots, q_{i|\Pi_i|}$

Initialize $\mathbf{q}_i = q_{i1}, \ldots, q_{i|\Pi_i|} = hotstarts()$
Initialize $l = 0, \mathbf{mq}_i = \mathbf{q}_i, \delta = \infty$
while $l < maxIter$ **do**
  $\mathbf{q}'_i = \mathbf{q}_i$
  $step = GeometricLineSearch(\mathbf{q}'_i, \Delta_{KL}^{\mathbf{q}'_i})$
  $\mathbf{q}_i = \mathbf{q}'_i - step * \Delta_{KL}^{\mathbf{q}'_i}$
  $\delta q_i = \Delta_{KL}^{\mathbf{q}_i} - \Delta_{KL}^{\mathbf{q}'_i}$
  if $\delta q_i < \delta$ **then**
    $\mathbf{mq}_i = \mathbf{q}_i$
    $\delta = \delta q_i$
  if $\delta q_i < t$ **then**
    break
  $l = l + 1$
return $\mathbf{mq}_i$

3.2 Pruning Rules

To find all near-optimal BNs given an approximating factor ϵ for a dataset I, we propose to compute two different sets of local scores for each node. The first set is the BIC scores when the conditional probability distributions for the candidate parents sets are represented by full CPTs. The second set is the BIC scores when the conditional probability distributions for the candidate parent sets are represented by noisy-OR relations. However, computing the local scores for all nodes is quite cost prohibitive—we would need a set of $n \cdot 2^{n-1}$ local scores for each of the two BIC scores. A solution is to prune the search space of candidate parent sets, provided that global optimality constraints of the full network structure are not violated. Adopting the terminology of Liao et al. (2019), we say that a candidate parent set Π_i can be *safely pruned* given a non-negative constant $\epsilon \in \mathbb{R}^+$ if Π_i cannot be the parent set of V_i in any network in the set of credible networks (see Definition 1). For computing BIC scores for full CPTs, we employ the following two pruning rules given by Liao et al. (2019) to find all near-optimal Bayesian networks given an approximating factor ϵ.

Lemma 2 *Given a node V_j, candidate parent sets Π_j and Π'_j, and some $\epsilon \in \mathbb{R}^+$, if $\Pi_j \subset \Pi'_j$ and $\sigma(\Pi_j) + \epsilon \leq \sigma(\Pi'_j)$, Π'_j can be safely pruned.*

Theorem 3 *Given a node V_j, candidate parent sets Π_j and Π'_j, and some $\epsilon \in \mathbb{R}^+$, if $\Pi_j \subset \Pi'_j$ and $\sigma(\Pi_j) - t(\Pi'_j) + \epsilon < 0$, Π'_j and all supersets of Π'_j can be safely pruned if σ is the BIC scoring function.*

For computing BIC scores for noisy-OR relations, we introduce two new pruning rules.

Lemma 4 *A candidate parent set Π_i of a node V_i that is consistently instantiated to zero throughout the dataset whenever the node is set to one can be safely pruned.*

Proof The candidate parent set Π_i cannot explain V_i in this configuration as there is no instance in the data file to indicate that Π_i affects the values of V_i. ■

Theorem 5 *Given a node V_j and some $\epsilon \in \mathbb{R}^+$, a candidate parent set Π_i with its penalty term greater than the sum of the score of the null parent set and ϵ can be safely pruned.*

Proof The null set is a subset of all candidate parent sets and by Lemma 2 any candidate parent set with a score exceeding the score of the null parent set can be safely pruned. Consider the definition of BIC for a parent set Π_i for node V_i, $\sigma(\Pi_i) = -L(\theta_i) + t(\Pi_i) \cdot w$. Let us have a candidate parent set with 2 or more parents, and with its penalty term greater than the score of the null parent set for V_i, $\sigma_i(\{\})$. Such a parent set will score lower than the null parent set as log-likelihood is negative and can be safely pruned; i.e., $t(\Pi_i) \cdot w > \sigma_i(\{\}) + \epsilon \Rightarrow -L(\theta_i) + t(\Pi_i) \cdot w > \sigma_i(\{\}) + \epsilon \Rightarrow \sigma(\Pi_i) > \sigma_i(\{\}) + \epsilon$. ■

3.3 Algorithm for ϵBNSL

Here we give our overall algorithm for ϵBNSL, a principled way to automatically select between full CPTs and noisy-OR relations, given a dataset and an approximation factor ϵ.

- **Step 1.** Determine the BIC scores when fitting a full CPT for all candidate parent sets that could not be pruned with pruning rules from (Liao et al., 2019) using Equation 1.
- **Step 2.** Determine the BIC scores when fitting a noisy-OR relation for all candidate parent sets that could not be pruned using our pruning rules in Section 3.2. Here, the noisy-OR parameters are fit using Algorithm 1, which minimizes the KL divergence between the full-CPT and the noisy-OR CPT. These parameters are used to compute the noisy-OR BIC score.
- **Step 3.** Merge these two score sets, using pruning rules Lemma 2 and Theorem 3, into a list of scores for candidate parent sets for each node in the dataset. During merging scores of a node V_j, we have to examine only cases where a candidate parent set Π_j belongs to the set of BIC scores and its superset Π'_j belongs to the noisy-OR BIC scores and vice versa.
- **Step 4.** The scores obtained in Step 3 are used to learn the set of credible networks using a developmental version of GOBNILP (Cussens and Bartlett, 2012), $gobnilp\_dev$ (Liao et al., 2019), which can be used to solve the ϵBNSL problem and collect all the networks in the credible set for a given approximation factor.

4. Experimental Evaluation

In this section, we show the accuracy of Algorithm 1[1] in computing the noisy-OR parameters for synthetic BNs with embedded noisy-OR relations. We also show significant presence of noisy-OR relations in standard benchmark networks. Finally, we test the performance of our learned networks against ground truth networks. All experiments are conducted on computers with 2.2 GHz Intel E7-4850V3 CPUs. Each experiment is limited to 64 GB of memory and 24 hours of CPU time.

1. Code available at https://github.com/CharupriyaSharma/eBNSLNoisyOR

Parent	$N = 100$		$N = 500$		$N = 1000$	
Size	KL	EM	KL	EM	KL	EM
2	0.16	1.07	0.07	1.07	0.05	1.00
3	0.21	1.18	0.09	1.11	0.07	1.07
4	0.27	1.04	0.11	1.27	0.07	1.26
5	0.25	1.50	0.11	1.54	0.08	1.57
6	0.34	1.99	0.16	2.04	0.10	2.06
7	0.41	2.09	0.24	2.03	0.16	1.99

Parent	$N = 100$		$N = 500$		$N = 1000$	
Size	KL	EM	KL	EM	KL	EM
2	0.04	0.05	0.01	0.01	0.00	0.01
3	0.13	0.32	0.02	0.05	0.01	0.03
4	0.33	1.32	0.06	0.24	0.03	0.12
5	1.02	4.91	0.18	1.20	0.07	0.55
6	1.33	9.02	0.33	3.48	0.16	2.06
7	2.54	12.08	1.07	11.68	0.60	6.55

Table 1: (Left) Median relative error in noisy-OR parameters and (right) median conditional KL divergence of noisy-OR CPTs learned by Algorithm 1, denoted KL, and the expectation-maximization algorithm, denoted EM, from ground truth for various parent set sizes.

4.1 Recovery of Noisy-ORs in Synthetic Datasets

To evaluate the accuracy of Algorithm 1 in finding the noisy-OR parameters and minimizing conditional KL divergence, we used synthetic BNs which consisted of a single noisy-OR. The parent set sizes were in the range $\{2, \ldots, 7\}$, all parent nodes had priors of 0.5, and the noisy-OR parameters $\mathbf{q} = q_1, \ldots, q_{|\Pi|}$ in the ground truth were uniformly sampled from the set $\{0.01, 0.02, \ldots, 0.99\}$. Thirty tests were performed at each parent set size.

We randomly generated datasets from the synthetic BNs with 100, 500 and 1000 instances, respectively. Algorithm 1 was applied to a dataset and the noisy-OR parameters estimated by the algorithm were compared against the parameters in the ground truth network (see Table 1). As well, the conditional KL divergence was computed between the noisy-OR CPT for the estimated parameters and the noisy-OR CPT for the ground truth parameters (see Equation 2). We also compared our results against the expectation-maximization algorithm for noisy-OR proposed by Vomlel (2006), the code for which was supplied by the author. As shown in Table 1, Algorithm 1 estimated the ground truth parameters with significantly higher accuracy than the EM algorithm. Algorithm 1 also had much lower conditional KL divergence.

4.2 Experiments on Standard Benchmarks: Presence of Noisy-OR Relations

To evaluate the ability of our overall algorithm for ϵBNSL (see Section 3.3) to learn networks with noisy-OR relations, we used standard datasets from the UCI Machine Learning Repository (https://archive.ics.uci.edu/). The datasets used were all binary or made binary.

The overall algorithm for ϵBNSL was applied to a dataset to learn the set of credible networks using a Bayes Factor, $BF = 20$. Out of the 13 (in a total of 16) benchmarks the algorithm was able to solve, 9 benchmarks showed a presence of noisy-OR relations (see Table 2). Specifically, these 9 benchmarks had 2 or more nodes that were assigned noisy-OR relations in at least 28% of the networks in the credible set. Also, 7 benchmarks had at least one node that was assigned a noisy-OR relation in all of the networks in the credible set. Note that some benchmarks, such as hepatitis and parkinsons, select noisy-OR relations for around half of their nodes, which shows that using only full CPTs could have resulted in overfitting. Further, optimal BNs containing noisy-OR relations were consistently found to have better scores than that of optimal networks found using only full CPTs. We also examined the effectiveness of the pruning rules (Steps 2 and 3 of the algorithm).

Dataset	n	N	nodes	ave.	max.	Dataset	n	N	nodes	ave.	max.
adult	14	32,561	0	0.0	0.0	autos	26	159	13	76.0	100.0
nltcs	16	3,236	0	0.0	0.0	horse	28	300	3	97.4	100.0
msnbc	17	58,265	0	0.0	0.0	flag	29	194	10	77.6	100.0
zoo	17	101	7	41.9	99.4	wdbc	31	569	OT	OT	OT
letter	17	20,000	OT	OT	OT	soybean	36	266	9	86.1	100.0
hepatitis	20	155	10	76.9	100.0	alarm	37	1,000	2	28.8	56.4
parkinsons	23	195	11	51.2	100.0	bands	37	277	8	63.4	100.0
sensors	25	5,456	OT	OT	OT	spectf	45	267	0	0.0	0.0

Table 2: Total number of nodes where a noisy-OR relation is selected (nodes) and average (ave.) and maximum (max.) percentage of networks in the set of credible networks that select noisy-OR relations for these nodes, for various benchmarks with n nodes and N instances in the dataset. OT indicates a dataset that could not be solved within the time limit.

On these benchmarks, the rules safely pruned away from 89.17% to 99.99% of the candidate parent sets, showing that the pruning rules are highly effective.

4.3 Performance on Ground Truth Networks

To further evaluate our overall algorithm for ϵBNSL (see Section 3.3), we used real-world Bayesian networks from the Bayesian Network Repository (www.bnlearn.com/bnrepository). The variables in the networks were made binary and their corresponding CPTs compressed (see Table 3; BNs without a _b suffix were already binary). From each ground truth network, we randomly generated datasets with 100, 500, and 1000 samples. We then ran our structure learning algorithm on the datasets to learn the set of credible networks, fixed the CPT parameters using maximum likelihood estimation and measured relative inference error against the ground truth network.

Table 3 shows the median relative inference error of the best scoring and the worst scoring networks in the set of credible networks, as well as that of the best-scoring network with full CPTs (i.e., not containing noisy-OR relations), against that of the ground truth network. Overall the inference error of the best scoring network is comparable to that of the full CPT. Somewhat surprisingly, the error for the worst scoring network can be smaller than for the best scoring network or the full CPT.

To perform inference on our learned set of credible networks, we generated evidence for 10% of nodes in the network. The nodes were randomly selected. For one trial, we selected a state of every node in the evidence, which was set according to the node's posterior probability distribution in the model, conditional on the evidence observed up till this point. Then, we computed the posterior probability distributions over the non-evidence nodes for our learned network and for the ground truth network. The inference errors were the differences between these values. We repeated the described procedure 1000 times for each of the networks. Inference was performed using JavaBayes (www.cs.cmu.edu/~javabayes), which was extended to take in an evidence file and two BNs for comparison. Our results our consistent with Zagorecki and Druzdzel (2013), who show that in three real-world Bayesian networks, noisy-OR/MAX relations were a good fit for up to 50% of the CPTs in these networks and that converting some CPTs to noisy-OR/MAX relations gave good approximations when answering probabilistic queries.

Bayesian network	n	$N = 100$ best	$N = 100$ worst	$N = 100$ CPT	$N = 500$ best	$N = 500$ worst	$N = 500$ CPT	$N = 1,000$ best	$N = 1,000$ worst	$N = 1,000$ CPT
earthquake	5	0.03	0.91	0.00	0.02	0.98	0.00	0.26	0.53	1.00
survey_b	6	0.05	0.69	0.00	0.02	0.74	0.00	0.01	0.75	0.00
asia	8	0.04	0.13	0.92	0.04	0.92	0.02	0.02	0.90	0.08
sachs_b	11	0.43	0.68	0.18	0.70	0.60	0.21	0.68	0.62	0.01
child_b	20	0.05	0.91	0.01	0.05	0.88	0.07	0.05	0.85	0.04
insurance_b	27	0.67	0.72	0.70	0.65	0.71	0.68	0.65	0.68	0.68
alarm_b	37	0.04	0.99	0.01	0.08	0.99	0.05	0.05	OT	0.06

Table 3: Median relative inference error for the best and worst scoring network in the set of credible networks learned by Algorithm 1 and the full CPT against the ground truth network. The datasets with N instances were generated from various ground truth BNs with n nodes.

5. Conclusion

Existing successful approaches for learning Bayesian networks from data use the well-known score-and-search approach. We extend the score-and-search approach to simultaneously learn the best global structure and the best local structure when the choice is either a full CPT or a noisy-OR relation for a candidate parent set of a node in the network. We show how to score a causal noisy-OR relation for a candidate parent set by fitting the best possible noisy-OR to the data, and we show how to effectively prune the search space while maintaining the optimality of the networks that are learned. Our experimental results provide evidence of the effectiveness of our approach. In particular, it was found that noisy-OR relations appeared in a significant proportion of the learned networks, for well known datasets.

References

M. Bartlett and J. Cussens. Advances in Bayesian network learning using integer programming. In *Proc. of UAI*, pages 182–191, 2013.

C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller. Context-specific independence in Bayesian networks. In *Proc. of UAI*, pages 115–123, 1996.

D. M. Chickering, D. Heckerman, and C. Meek. A Bayesian approach to learning Bayesian networks with local structure. In *Proc. of UAI*, pages 80–89, 1997.

J. Cussens and M. Bartlett. GOBNILP 1.2 user/developer manual. *University of York, York*, 2012.

A. Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge Univ. Press, 2009.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. of the Royal Statistical Society Series B*, 39:1–38, 1977.

F. J. Díez. Parameter adjustment in Bayes networks. The generalized noisy OR-gate. In *Proc. of UAI*, pages 99–105, 1993.

N. Friedman and M. Goldszmidt. Learning Bayesian networks with local structure. In *Learning in graphical models*, pages 421–459. Springer, 1998.

I. J. Good. A causal calculus. *The British J. for the Philosophy of Science*, 12(45):43–51, 1961.

M. Henrion. Some practical issues in constructing belief networks. In *Proc. of UAI*, pages 132–139, 1987.

R. E. Kass and A. E. Raftery. Bayes factors. *J. of the Amer. Stat. Assoc.*, 90(430):773–795, 1995.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, 2009.

W. Lam and F. Bacchus. Using new data to refine a Bayesian network. In *Proc. of UAI*, pages 383–390, 1994.

Z. A. Liao, C. Sharma, J. Cussens, and P. van Beek. Finding all Bayesian network structures within a factor of optimal. In *Proc. of AAAI*, pages 7892–7899, 2019.

K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

D. Poole and N. L. Zhang. Exploiting contextual independence in probabilistic inference. *J. of AI Research*, 18:263–313, 2003.

G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464, 1978.

T. Talvitie, R. Eggeling, and M. Koivisto. Learning Bayesian networks with local structure, mixed variables, and exact algorithms. *Int'l J. of Approximate Reasoning*, 115:69–95, 2019.

P. van Beek and H.-F. Hoffmann. Machine learning of Bayesian networks using constraint programming. In *Proc. of CP*, pages 428–444, 2015.

J. Vomlel. Noisy-OR classifier. *Int'l J. of Intelligent Systems*, pages 381–398, 2006.

Y. Xiang. Direct causal structure extraction from pairwise interaction patterns in NAT modeling Bayesian networks. *Int'l J. of Approximate Reasoning*, 105:175–193, 2019.

Y. Xiang and B. Baird. Compressing Bayesian networks: Swarm-based descent, efficiency, and posterior accuracy. In *Proc. of CAI*, pages 3–16. Springer, 2018.

C. Yuan, B. Malone, and X. Wu. Learning optimal Bayesian networks using A* search. In *Proc. of IJCAI*, pages 2186–2191, 2011.

A. Zagorecki and M. J. Druzdzel. Knowledge engineering for Bayesian networks: How common are noisy-MAX distributions in practice? *IEEE Trans. on SMC*, 43(1):186–195, 2013.

N. L. Zhang and D. Poole. Exploiting causal independence in Bayesian network inference. *J. of AI Research*, 5:301–328, 1996.