On Pruning for Score-Based Bayesian Network Structure Learning

Alvaro H. C. Correia
Eindhoven University of Technology

James Cussens
University of York

Cassio P. de Campos
Eindhoven University of Technology

Abstract

Many algorithms for score-based Bayesian network structure learning (BNSL), in particular exact ones, take as input a collection of potentially optimal parent sets for each variable in the data. Constructing such collections naively is computationally intensive since the number of parent sets grows exponentially with the number of variables. Thus, pruning techniques are not only desirable but essential. While good pruning rules exist for the Bayesian Information Criterion (BIC), current results for the Bayesian Dirichlet equivalent uniform (BDeu) score reduce the search space very modestly, hampering the use of the (often preferred) BDeu. We derive new non-trivial theoretical upper bounds for the BDeu score that considerably improve on the state-of-the-art. Since the new bounds are mathematically proven to be tighter than previous ones and at little extra computational cost, they are a promising addition to BNSL methods.

1 Introduction

A Bayesian network (Pearl, 1988) is a widely used probabilistic graphical model. It is composed of (i) a *structure* defined by a directed acyclic graph (DAG) where each node is associated with a random variable, and where arcs represent probabilistic dependencies entailing the *Markov* condition: every variable is conditionally independent of its non-descendant variables given its parents; and (ii) a collection of conditional probability distributions defined for each variable given its parents in the graph. Their graphical nature make Bayesian networks ideal for complex probabilistic relationships existing in many real-world problems (Cussens et al., 2013).

Bayesian network structure learning (BNSL) is essentially a search problem where we aim at finding the optimal graph given some data. We assume discrete data and tackle score-based learning, that is, we define the optimal graph as the structure maximising a data-dependent score (Heckerman et al., 1995). In particular, we focus on the *Bayesian Dirichlet equivalent uniform* (BDeu) score (Cooper and Herskovits, 1992), which consists in the log probability of the graph given (multinomial) data and a uniform prior on structures. The BDeu score is decomposable; namely we can write it as a sum of *local scores* of the domain variables:

$$\mathrm{BDeu}(\mathcal{G}) = \sum_{i \in V} \mathrm{LBDeu}(i, S_i),$$

where LBDeu is the local score function; $V = \{1, \ldots, n\}$ is the set of (indices of) variables in the data, which are in correspondence with nodes of the network to be learned; and $S_i \subseteq V^{\setminus i}$, with $V^{\setminus i} = V \setminus \{i\}$, the parent set of node i in the DAG $\mathcal{G}$.

A common approach to *exact* BNSL divides the problem into two steps:

1. Candidate Parent Set Identification: For each variable, find a suitable collection of candidate parent sets and their local scores.
2. Structure Optimisation: Given the collection of candidate parent sets, choose a parent set for each variable so as to maximise the overall score while avoiding directed cycles.

This paper concerns pruning ideas to help solve candidate parent set identification. Simply put, we aim at reducing the number of BDeu scores we have to compute by discarding parent sets that will not lead to an optimal solution at the second step.

BNSL is known to be NP-hard (Chickering et al., 2004) and the subproblem of parent set identification is unlikely to admit a polynomial-time (in n) algorithm; it is proven to be LOGSNP-Hard for BIC (Koivisto, 2006). As a compromise, one typically chooses a maximum in-degree d (number of parents per node) and computes the score only for parent sets with in-degree at most d.

Proceedings of the 23$^{\mathrm{rd}}$International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

Naturally, that does reduce the search space but comes at the cost of discarding numerous potentially optimal graphs. Conversely, increasing the maximum in-degree can considerably improve the chances of finding better structures but requires higher computing time: there are $\Theta(n^d)$ candidate parent sets (per variable) if an exhaustive search is performed with in-degree d, and 2^{n-1} without any in-degree constraint. The large search space is an important limiting factor in BNSL, as $d > 2$ is already prohibitively expensive for many interesting applications (Bartlett and Cussens, 2017).

Our goal is then to prune this search space more aggressively to help scale exact BNSL with BDeu. We provide new theoretical upper bounds for the local scores that allow us to identify and discard non-optimal parent sets without ever having to compute their scores. These new upper bounds are efficient and can be readily integrated to any searching approach (Chen et al., 2016; Cussens, 2011; de Campos and Ji, 2011; de Campos et al., 2009; Jaakkola et al., 2010; Koivisto and Sood, 2004; Yuan and Malone, 2012, 2013). While our study has been motivated by the scientific interest in solving the BNSL problem in an exact manner, we shall note that local scores for a variable given its parents also have a probabilistic interpretation (the decomposition of the score comes from independence assumptions). Therefore, new approaches to prune such search space of parent sets can be useful for other purposes too.

The paper is organised as follows. Section 2 provides the notation and required definitions, as well as a brief description of the current best bound for BDeu in the literature, which we call ub_f. Section 4 presents a new improved bound ub_g whose derivation follows the same mathematical approach as the existing state-of-the-art bound but further exploits properties of the score function to get better results. This new bound is provably tighter than previous ones but still does not capture all cases and other bounds can be devised. Section 5 looks at the problem from a new angle and introduces a bound ub_h based on a (tweaked) maximum likelihood estimation. Bounds ub_g and ub_h leverage different aspects of the problem and we show how they can be effectively combined for an even more aggressive pruning in Section 6. Finally, Section 7 concludes the paper and gives directions for future research.

2 Definitions and Notation

First of all, since the collection of scores are computed independently for each variable in the dataset (BDeu is decomposable), we drop i from the notation and use simply $\mathrm{LBDeu}(S)$ to refer to the score of node i with parent set $S \subseteq V^{\setminus i}$. We need some further notation:

- The state space of variable i is denoted by $c(i)$. Similarly, $c(S)$ is the set of all joint instantiations of the random variables in $S \subseteq V$, that is, $c(S)$ is the Cartesian product of the state space of involved variables, $c(S) = \times_{i \in S} c(i)$. We denote the size of state space $c(S)$ as $q(S) = |c(S)|$, and we abuse notation to say $q(i) = |c(i)|$.
- We reserve i for (indices of) variables and j for instances of a state space, e.g., $j_S \in c(S)$. The subscript is omitted if clear from the context.
- The data $\mathcal{D}$ is a *multiset* (repetitions are allowed) of elements from $c(V)$, with $\mathcal{D}^S$ the projection of $\mathcal{D}$ onto variables $S \subseteq V$ (note that $\mathcal{D} = \mathcal{D}^V$). The same notation applies for projections of instantiations, e.g. j^S. Moreover, we use $\mathcal{D}^S(j_{S'}) \subseteq \mathcal{D}^S$ to denote the elements of $\mathcal{D}^S$ compatible with a given $j_{S'} \in c(S')$, that is, $\mathcal{D}^S(j_{S'}) = \{j_S : j_S \in \mathcal{D}^S, j_S^{S\cap S'} = j_{S'}^{S\cap S'}\}$. Finally, we use $\mathcal{D}_u$ instead of $\mathcal{D}$ to denote the set of unique elements from a given multiset $\mathcal{D}$.
- For $j \in c(S)$, we define $n_j = |\mathcal{D}^S(j)|$, that is, the number of occurrences of j in $\mathcal{D}^S$.
- The vector $\vec{\alpha}_j = (\alpha_{j,k})_{k \in c(i)}$ is the prior for parent set $S \subseteq V^{\setminus i}$ under configuration $j \in c(S)$. In the BDeu score, $\vec{\alpha}_j$ satisfies $\alpha_{j,k} = \alpha_{\mathrm{ess}}/q(S \cup \{i\})$, where α_{ess} is the equivalent sample size, a pre-defined user parameter to define the strength of the prior.

Let $\Gamma_\alpha(x) = \frac{\Gamma(x+\alpha)}{\Gamma(\alpha)}$ for x non-negative integer and $\alpha > 0$ (Γ denotes the Gamma function). Denote $\sum_{k \in c(i)} \alpha_{j,k} = \alpha_{\mathrm{ess}}/q(S)$ by α_j. The local score for i with parent set $S \subseteq V^{\setminus i}$ can be written as

$$\mathrm{LBDeu}(S) = \sum_{j \in c(S)} \mathrm{LLBDeu}(S, j), \text{ and}$$

$$\mathrm{LLBDeu}(S, j) = -\log \Gamma_{\alpha_j}(n_j) + \sum_{k \in c(i)} \log \Gamma_{\alpha_{j,k}}(n_{j,k}) \,.$$

$\mathrm{LBDeu}(S)$ is a sum of $q(S)$ values each of which is specific to a particular instantiation of variables in S. We call such values *local local BDeu scores (llB)*. In particular, $\mathrm{LLBDeu}(S, j) = 0$ if $n_j = 0$, so we concentrate on instantiations j that do appear in the data:

$$\mathrm{LBDeu}(S) = \sum_{j \in \mathcal{D}_u^S} \mathrm{LLBDeu}(S, j) \,.$$

This formula does not come by chance. In Section 5 we discuss its relation with the posterior probability of having S as parent of i.

3 Pruning in Candidate Parent Set Identification

The pruning of parent sets rests on the (simple) observation that a parent set cannot be optimal if one of its subsets has a higher score (Teyssier and Koller, 2005). Thus, when learning Bayesian networks from data using BDeu, it is useful to have an upper bound

$$\mathrm{ub}(S) \geqslant \max_{T:T\supset S} \mathrm{LBDeu}(T) \tag{1}$$

so as to potentially prune a whole area of the search space at once. Ideally, one would like an upper bound that is both tight (with respect to the inequality in Expression 1) and cheap to compute, so that one can score parent sets incrementally, and at the same time check whether it is worth 'expanding' them: if $\mathrm{ub}(S)$ is not greater than $\max_{R:R\subseteq S}\mathrm{LBDeu}(R)$, then it is unnecessary to expand S. Figure 1 illustrates how a hypothetical bound would prune the search space.

With that in mind, we can define candidate parent set identification more formally.

Definition 1 (Candidate Parent Set Identification)**.** *For each variable $i \in V$, find a collection of parent sets*

$$\mathcal{L}_i = \{S \subseteq V^{\setminus i} : S' \subset S \Rightarrow \mathrm{LBDeu}(S') < \mathrm{LBDeu}(S)\}\,.$$

Unfortunately, we cannot predict the elements of $\mathcal{L}_i$ and have to compute the scores for a list L_i potentially much larger than $\mathcal{L}_i$. The practical benefit of our bounds is to reduce $|L_i|$, thus lowering the computational cost of BNSL, while ensuring we do not miss any potentially optimal parent set, that is, $L_i \supseteq \mathcal{L}_i$. Before presenting the current best bound in the literature (Cussens, 2012; de Campos and Ji, 2010, 2011), we give a lemma on the variation of counts with expansions of parent sets.

Lemma 1. *For $S \subseteq T \subseteq V^{\setminus i}$, $j_S \in \mathcal{D}_u^S$ and $j_T \in \mathcal{D}_u^T$ with $j_T^S = j_S$, we have $|\mathcal{D}_u^{T\cup\{i\}}| \geqslant |\mathcal{D}_u^{S\cup\{i\}}|$, and $|\mathcal{D}_u^{T\cup\{i\}}(j_T)| \leqslant |\mathcal{D}_u^{S\cup\{i\}}(j_S)|$.*

Proof. Given that $S \subseteq T \subseteq V^{\setminus i}$, every instantiation in $\mathcal{D}_u^{S\cup i}$ is compatible with one or more elements of $\mathcal{D}_u^{T\cup i}$, and thus $|\mathcal{D}_u^{T\cup i}| \geqslant |\mathcal{D}_u^{S\cup i}|$. The relationship is reversed when we consider unique occurrences compatible with a given instantiation. By construction $j_T^S = j_S$, so if there is an instantiation $j_T \in \mathcal{D}_u^T$, there must be at least one corresponding $j_S \in \mathcal{D}_u^S$, and it follows that $|\mathcal{D}_u^{T\cup\{i\}}(j_T)| \leqslant |\mathcal{D}_u^{S\cup\{i\}}(j_S)|$. Note that both $|\mathcal{D}_u^{T\cup\{i\}}(j_T)|$ and $|\mathcal{D}_u^{S\cup\{i\}}(j_S)|$ are bounded by $q(i)$: one instantiation for each value child i can assume. □

As an example, consider the small dataset of Table 1. The number of non-zero counts never decreases as we add a new variable to the parent set of variable $i = 3$. With $S = \{1\}$ and $T = \{1, 2\}$, we have $|\mathcal{D}_u^{S\cup\{i\}}| = 3$ and $|\mathcal{D}_u^{T\cup\{i\}}| = 4$. Conversely, the number of (unique) occurrences compatible with a given instantiation of the parent set never increases with its expansion: for example with $j_S = (1)$ and $j_T = (1, 1)$, we have $|\mathcal{D}_u^{S\cup\{i\}}(j_S)| = 2$ and $|\mathcal{D}_u^{T\cup\{i\}}(j_T)| = 2$.

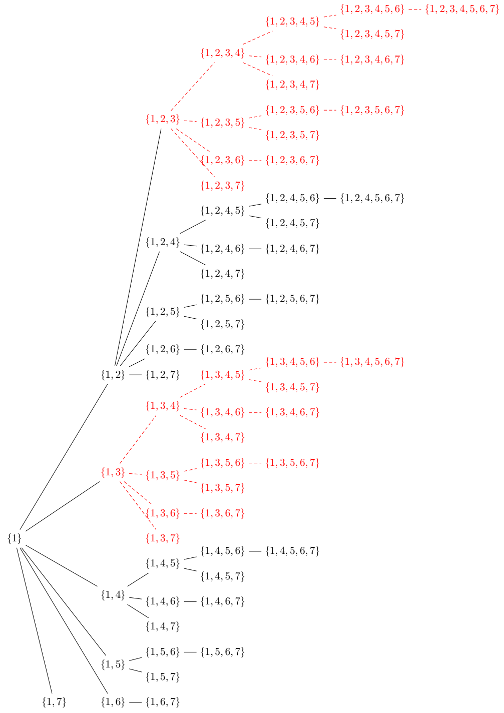

Figure 1: Illustration of potential parent sets in a dataset with 8 variables (the 8^{th} one is the child and does not show). This is still a small part of the search space with only sets including variable 1. In red dashed lines, the sets pruned if $\mathrm{LBDeu}(\{1\}) \geqslant \mathrm{ub}(\{1, 3\})$.

We now introduce function f that, for a variable i, is defined on the sets of potential parents $S \subseteq V^{\setminus i}$, and observed instantiations $j \in \mathcal{D}_u^S$:

$$\begin{aligned} f(S, j) &= -|\mathcal{D}_u^{S\cup\{i\}}(j)| \log q(i)\,, \\ f(S) &= \sum_{j\in\mathcal{D}_u^S} f(S, j). \end{aligned} \tag{2}$$

Theorem 1 (ub_f)**.** *For a variable i, a potential parent set $S \subseteq V^{\setminus i}$ and its instantiations $j \in \mathcal{D}_u^S$, we have that* $\mathrm{LLBDeu}(S, j) \leqslant f(S, j)$.

Moreover, if $\mathrm{LBDeu}(S') \geqslant \sum_{j\in\mathcal{D}_u^S} f(S, j) = f(S)$ *for some $S' \subset S$, then all $T \supseteq S$ are not in $\mathcal{L}_i$ (Cussens and Bartlett, 2015; de Campos and Ji, 2011).*

From Theorem 1, we get an upper bound on the local BDeu score of all supersets of parent set S

$$\mathrm{ub}_f(S) = f(S) \geqslant \max_{T:T\supset S} \mathrm{LBDeu}(T). \tag{3}$$

In words, we compute the number of non-zero counts per instantiation, $|\mathcal{D}_u^{S\cup\{i\}}(j)|$, and we 'gain' $\log q(i)$ for each of them. Note that $f(S) = -|\mathcal{D}_u^{S\cup\{i\}}| \log q(i)$, which by Lemma 1 is monotonically non-increasing over expansions of the parent set S. Hence $f(S)$ is not only

Table 1: Example of data $\mathcal{D}$, its reductions by parent sets $S = \{1\}$ and $T = \{1, 2\}$, and the number of unique occurrences compatible with $j_S \in \mathcal{D}_u^S$ and $j_T, j'_T \in \mathcal{D}_u^T$, with $j_T^S = {j'}_T^S = j_S$. The child variable is $i = 3$, and we have $j_S = (1)$, $j_T = (1, 1)$, $j'_T = (1, 0)$.

$\mathcal{D}$

1	2	3
0	0	0
1	0	0
1	1	0
1	1	1

$\mathcal{D}_u^{S\cup\{i\}}$

1	3
0	0
1	0
1	1

$\mathcal{D}_u^{T\cup\{i\}}$

1	2	3
0	0	0
1	0	0
1	1	0
1	1	1

$\mathcal{D}_u^{S\cup\{i\}}(j_S)$

1	3
1	0
1	1

$\mathcal{D}_u^{T\cup\{i\}}(j_T)$

1	2	3
1	1	0
1	1	1

$\mathcal{D}_u^{T\cup\{i\}}(j'_T)$

1	2	3
1	0	0

an upper bound on LBDeu(S) but also on LBDeu(T) for every $T \supseteq S$. Bound ub_f is cheap to compute but is unfortunately too loose. We derive much tighter upper bounds on LLBDeu(S, j) (where $n_j > 0$) by considering instantiation counts for the *full* parent set $V^{\setminus i}$, the parent set that includes all possible parents for child i. We call these *full instantiation counts.* Evidently, the number of full parent instantiations $q(V^{\setminus i})$ grows exponentially with $|V|$, but it is linear in $|\mathcal{D}|$ when we consider only the unique elements $\mathcal{D}_u^{V^{\setminus i}}$.

4 Exploiting the Gamma Function

First, we extend the current state-of-the-art upper bound of Theorem 1 by exploiting some properties of the Gamma function. For that, we need some intermediate results, where we assume $\alpha > 0$.

Lemma 2. *Let x be a positive integer. Then*

$$\Gamma_\alpha(0) = 1 \quad \textit{and} \quad \log \Gamma_\alpha(x) = \sum_{\ell=0}^{x-1} \log(\ell + \alpha)\,.$$

Proof. Follows from $\Gamma(x + 1) = x\Gamma(x)$. □

Lemma 3. *For x positive integer and $v \geqslant 1$,*

$$\log\left(\frac{\Gamma_\alpha(x)}{\Gamma_{\alpha/v}(x)}\right) \geqslant \log v\,.$$

Proof. By applying Lemma 2, we obtain

$$\sum_{\ell=0}^{x-1} \log \frac{\ell + \alpha}{\ell + \alpha/v} = \log v + \sum_{\ell=1}^{x-1} \log \frac{\ell + \alpha}{\ell + \alpha/v} \geqslant \log v\,,$$

as each term of the sum (if any) is greater than zero. □

Lemma 4. *Let x, y be non-negative integers such that $x + y > 0$. Then*

$$\begin{cases} \Gamma_\alpha(x + y) = \Gamma_\alpha(x)\,\Gamma_\alpha(y) & \textit{if } x \cdot y = 0\,, \\ \Gamma_\alpha(x + y) \geqslant \Gamma_\alpha(x)\,\Gamma_\alpha(y)\,(1 + y/\alpha) & \textit{otherwise.} \end{cases}$$

Proof. If x (resp. y) is zero, then $\Gamma_\alpha(x) = 1$ and the equality holds. Otherwise we apply Lemma 2 three times and manipulate the products:

$$\begin{aligned} \frac{\Gamma_\alpha(x + y)}{\Gamma_\alpha(x)\,\Gamma_\alpha(y)} &= \frac{\prod_{z=0}^{x+y-1}(z + \alpha)}{\prod_{z=0}^{x-1}(z + \alpha)\prod_{z=0}^{y-1}(z + \alpha)} \\ &= \prod_{z=y}^{x+y-1}(z + \alpha)\prod_{z=0}^{x-1}\frac{1}{(z + \alpha)} \\ &= \prod_{z=0}^{x-1}\frac{y + z + \alpha}{z + \alpha} \geqslant \frac{y + \alpha}{\alpha}\,, \end{aligned}$$

which holds since all terms in this final product are greater or equal to 1. □

Corollary 1. *Let $x_1, \ldots, x_k$ be a list of non-negative integers in decreasing order with $x_1 > 0$, then*

$$\Gamma_\alpha\left(\sum_{l=1}^{k} x_l\right) \geqslant \prod_{l=1}^{k}\Gamma_\alpha(x_l)\prod_{l=1}^{k'-1}(1 + x_l/\alpha)\,,$$

where $k' \leqslant k$ is the last positive integer in the list (in this notation, the second product on the right-hand side disappears if $k' = 1$).

Proof. Repeatedly apply Lemma 4 to $x_t + (\sum_{l=t}^{k} x_l)$ until all elements are processed. While both the current x_t and the rest of the list are positive (until $t = k' - 1$), we obtain the extra term $(1 + x_t/\alpha)$. After that, we only 'collect' the Gamma functions of the first product on the right-hand side, so the result follows. □

Lemma 5. *For $S \subseteq V^{\setminus i}$ and $j \in \mathcal{D}_u^S$, assume that $\vec{n}_j = (n_{j,k})_{k\in c(i)}$ are in decreasing order over $k = 1, \ldots, q(i)$ (this is without loss of generality, since we can name and process them in any order). Then for any $\alpha \geqslant \alpha_j = \alpha_{ess}/q(S)$, we have*

$$\begin{aligned} \mathrm{LLBDeu}(S, j) &\leqslant f(S, j) + g(S, j, \alpha), \\ g(S, j, \alpha) &= -\sum_{l=1}^{k'-1}\log(1 + n_{j,l}/\alpha)\,, \end{aligned}$$

where $k' \leqslant k$ is the largest index such that $n_{j,k'} > 0$.

Proof. First of all, we have

$$\begin{aligned} \mathrm{LLBDeu}(S, j) &= -\log \Gamma_{\alpha_j}(n_j) + \sum_{k\in c(i)} \log \Gamma_{\alpha_{j,k}}(n_{j,k}) \\ &= -\log \Gamma_{\alpha_j}\left(\sum_{k\in c(i)} n_{j,k}\right) + \sum_{k\in c(i)} \log \Gamma_{\alpha_{j,k}}(n_{j,k})\;. \end{aligned}$$

Since counts $n_{j,k}$ are in decreasing order by k, we apply Corollary 1:

$$\begin{aligned}
\mathrm{LLBDeu}(S,j) &\leqslant -\log\left(\prod_{l=1}^{q(i)}\Gamma_{\alpha_j}\left(n_{j,l}\right)\prod_{l=1}^{k'-1}(1+\frac{n_{j,l}}{\alpha_j})\right) \\
&\quad + \sum_{k\in c(i)}\log\Gamma_{\alpha_{j,k}}\left(n_{j,k}\right) \\
&= \sum_{k\in c(i)}\log\left(\frac{\Gamma_{\alpha_{j,k}}\left(n_{j,k}\right)}{\Gamma_{\alpha_j}\left(n_{j,k}\right)}\right) - \sum_{l=1}^{k'-1}\log\left(1+\frac{n_{j,l}}{\alpha_j}\right) \\
&\leqslant -|\mathcal{D}_u^{S\cup\{i\}}(j)|\log q(i) - \sum_{l=1}^{k'-1}\log\left(1+\frac{n_{j,l}}{\alpha}\right)
\end{aligned}$$

with $\alpha \geqslant \alpha_j$ and $\Gamma_{\alpha_{j,k}}\left(n_{j,k}\right)/\Gamma_{\alpha_j}\left(n_{j,k}\right) \leqslant -\log q(i)$ by Lemma 3 whenever $n_{j,k} > 0$. □

The difference here is the summation from the gap of the super-multiplicativity of Γ (Lemma 4 and Corollary 1). That extra term gives us a tighter bound on $\mathrm{LLBDeu}(S,j)$, but $g(S) = f(S) + \sum_{j\in\mathcal{D}_u^S} g(S,j,\alpha)$ is no longer monotonic over expansions of S (albeit monotone in α). Hence, $g(S)$ is not an upper bound on $\mathrm{LBDeu}(T)$ for every $T \supseteq S$, and we need further results on $g(S,j,\alpha)$.

Lemma 6. *For $S \subseteq T \subseteq V^{\setminus i}$, $j_T \in \mathcal{D}_u^T$, and $j_S \in \mathcal{D}_u^S$ with $j_T^S = j_S$, we have*

$$f(T,j_T) \geqslant f(S,j_S),$$

$$g(T,j_T,\alpha) \geqslant g(S,j_S,\alpha).$$

Proof. Because $j_T^S = j_S$, $|\mathcal{D}_u^{T\cup\{i\}}(j_T)| \leqslant |\mathcal{D}_u^{S\cup\{i\}}(j_S)|$. Moreover, $n_{j_T,k} \leqslant n_{j_S,k}$ for every $k \in c(i)$ (the counts get partitioned as more parents are introduced to arrive at T from S), so $(1+n_{j_T,k}/\alpha) \leqslant (1+n_{j_S,k}/\alpha)$ for every k, and the result follows. □

Using this property of g as described in Lemma 6, we can pick the best value of g over all full expansions j of a current instantiation j_S to create a valid bound:

Theorem 2 (ub_g). *Let $S \subseteq V^{\setminus i}$, $j_S \in \mathcal{D}_u^S$, Then*

$$\mathrm{LLBDeu}(S,j_S) \leqslant f(S,j_S) + \underline{g}(S,j_S)$$

$$\underline{g}(S,j_S) = \min_{j\in\mathcal{D}_u^{V^{\setminus i}}:\ j^S=j_S} g(V^{\setminus i},\ j,\ \alpha_{ess}/q(S))$$

Also, if $\mathrm{LBDeu}(S') \geqslant (f(S)+\sum_{j_S\in\mathcal{D}_u^S}\underline{g}(S,j_S)) = \underline{g}(S)$ *for some $S' \subset S$, then all $T \supseteq S$ are not in $\mathcal{L}_i$.*

Proof. First we prove that $f(S,j_S) + \underline{g}(S,j_S)$ is an upper bound for $\mathrm{LLBDeu}(S,j_S)$. From Lemma 6, if we take any instantiation of the fully expanded parent set, $j \in \mathcal{D}_u^{V^{\setminus i}} : j^S = j_S$, we have that $g(S,j_S,\alpha) \leqslant g(V^{\setminus i},j,\alpha)$ for any α. As Lemma 6 is valid for every full instantiation j, we take the minimum over them to get the tightest bound. From Lemma 5, $\mathrm{LLBDeu}(S,j_S) \leqslant f(S,j_S) + \underline{g}(S,j_S)$. Now, if we sum all the llBs, we obtain the second part of the theorem for S.

Finally, we need to show that this second part of the theorem holds for any $T \supset S$, which follows from $f(T) \leqslant f(S)$ (as the total number of non-zero counts only increases, by Lemma 1) and

$$\begin{aligned}
\sum_{j_T\in\mathcal{D}_u^T}\underline{g}(T,j_T) &= \sum_{j_S\in\mathcal{D}_u^S}\left(\sum_{j_T\in\mathcal{D}_u^T:\ j_T^S=j_S}\underline{g}(T,j_T)\right) \\
&\leqslant \sum_{j_S\in\mathcal{D}_u^S}\underline{g}(S,j_S)\,.
\end{aligned}$$

That holds as $\underline{g}(T,j_T) \leqslant 0$ and, with $j_T^S = j_S$, at least one term $\underline{g}(T,j_T)$ is smaller than $\underline{g}(S,j_S)$, as their minimisation spans the same full instantiations (and $g(\cdot,\cdot,\alpha)$ is non-decreasing on α). □

In brief, the relevance of Theorem 2 is that it gives us a tighter upper bound $\mathrm{ub}_g(S) \leqslant \mathrm{ub}_f(S)$, such that

$$\begin{aligned}
\mathrm{ub}_g(S) = \underline{g}(S) &= (f(S) + \sum_{j_S\in\mathcal{D}_u^S}\underline{g}(S,j_S)) \\
&\geqslant \max_{T:T\supset S}\mathrm{LBDeu}(T).
\end{aligned}$$

Therefore, this bound is always equal or superior to the current state-of-the-art bound. Moreover, the overhead of computing the bounds is negligible if a smart implementation is used (one that reuses computations that are nevertheless required for calculating the scores). The process which constructs contingency tables of counts for local score computations (say, from an AD-tree) is the main bottleneck in scoring, but it can be cheaply extended to simultaneously produce tables of sets of "full instantiations" for the computation of upper bounds where, for instance, addition of counts are replaced with unions of sets. While this technical detail is irrelevant for the mathematical proofs here, it is important to point out that the new bounds imply very little extra computational costs.

5 Exploiting the Likelihood Function

Bound ub_g of previous section was based on the best full instantiation $j \in \mathcal{D}_u^{V^{\setminus i}}$ that is compatible with an llB of the parent set S. Knowing that function g is monotonic over parent set sizes, we could look at an instantiation of the fully extended parent set to derive a bound for the llB of S and all its supersets. Even though the results are valid for every full instantiation, we can only compute bound ub_g using one of them at a time. The new bound of this section comes from the realisation that it is possible to exploit all full instantiations to derive a valid bound on the llB of S. For that purpose, we need some properties of inferences with the Dirichlet-multinomial distribution and conjugacy.

The BDeu score is simply the log marginal probability of the observed data given suitably chosen Dirichlet priors over the parameters of a BN structure. Consequently, llBs are intimately connected to the Dirichlet-multinomial conjugacy. Given a Dirichlet prior $\vec{\alpha}_j = (\alpha_{j,1}, \ldots, \alpha_{j,q(i)})$, the probability of observing data $\mathcal{D}_{\vec{n}_j}$ with counts $\vec{n}_j = (n_{j,1}, \ldots, n_{j,q(i)})$ is:

$$\log \Pr(\mathcal{D}_{\vec{n}_j}|\vec{\alpha}_j) = \log \int_p \Pr(\mathcal{D}_{\vec{n}_j}|p)\Pr(p|\vec{\alpha}_j)dp\,,$$

where the first distribution under the integral is multinomial and the second is Dirichlet. Note that

$$\log \int_p \Pr(\mathcal{D}_{\vec{n}_j}|p)\Pr(p|\vec{\alpha}_j)dp \leqslant \max_p \log \Pr(\mathcal{D}_{\vec{n}_j}|p), \quad (4)$$

since $\int_p \Pr(p|\vec{\alpha}_j)dp = 1$. Note also that llBs are not the probability of observing sufficient statistics counts, but of a particular dataset, that is, there is no multinomial coefficient which would consider all the permutations yielding the same sufficient statistics. Therefore, we may devise a new upper bound based on the maximum (log-)likelihood estimation.

Lemma 7. *Let $S \subseteq V^{\setminus i}$ and $j \in \mathcal{D}_u^S$. Then* LLBDeu(S,j) $\leqslant$ $ML(\vec{n}_j)$, *where we have that* $ML(\vec{n}_j) = \sum_{k\in c(i)} n_{j,k}\log(n_{j,k}/n_j)$. *(In this notation, we use $0 \log 0 = 0$.)*

Proof. The llB is simply the log probability of observing a data sequence with counts $\vec{n}_j$ under a Dirichlet-multinomial distribution with parameter vector $\vec{\alpha}_j$. The result follows from Expression (4) and holds for any prior $\vec{\alpha}_j$. □

Corollary 2. *Let $S \subseteq V^{\setminus i}$ and $j_S \in \mathcal{D}_u^S$. Then* LLBDeu$(S, j_S) \leqslant \sum_{j\in\mathcal{D}_u^{V^{\setminus i}}:\ j^S=j_S} ML(\vec{n}_j)$.

Proof. This follows from the properties of the maximum likelihood estimation, because it is monotonically non-decreasing with the expansion of parent sets (in terms of maximum likelihood, we fit the distribution just as well or better when having more parents). □

We can improve further on this bound of Corollary 2 by considering llBs as a function h of α for fixed $\vec{n}_j$, since we can study and exploit the shape of their curves. We define

$$h_{\vec{n}_j}(\alpha) = -\log\Gamma_\alpha(n_j) + \sum_{k\in c(i)} \log\Gamma_{\alpha/q(i)}(n_{j,k})\,.$$

Lemma 8. *If $\nexists k : n_{j,k} = n_j$, then $h_{\vec{n}_j}$ is a concave function for positive $\alpha \leqslant 1$.*

Proof. (This result can also be obtained from (Levin and Reeds, 1977).) Using the identity in Lemma 2, or, equivalently, by exploiting known properties of the digamma and trigamma functions, we have

$$\frac{\partial h_{\vec{n}_j}}{\partial\alpha}(\alpha) = -\sum_{\ell=0}^{n_j-1}\frac{1}{\ell+\alpha} + \sum_{k=1}^{q(i)}\sum_{\ell=0}^{n_{j,k}-1}\frac{1}{\ell q(i)+\alpha}, \text{ and}$$

$$\frac{\partial^2 h_{\vec{n}_j}}{\partial\alpha^2}(\alpha) = \sum_{\ell=0}^{n_j-1}\frac{1}{(\ell+\alpha)^2} - \sum_{k=1}^{q(i)}\sum_{\ell=0}^{n_{j,k}-1}\frac{1}{(\ell q(i)+\alpha)^2}\,.$$

It suffices to show that $\frac{\partial^2 h_{\vec{n}_j}}{\partial\alpha^2}(\alpha)$ is always negative under the conditions of the theorem. If there are at least two $n_{j,k} > 0$, then

$$\frac{\partial^2 h_{\vec{n}_j}}{\partial\alpha^2}(\alpha) \leqslant \sum_{\ell=0}^{n_j-1}\frac{1}{(\ell+\alpha)^2} - \frac{2}{\alpha^2}$$

simply by ignoring all those negative terms with $\ell \geqslant 1$. Now we approximate it by the infinite sum of quadratic reciprocals:

$$\begin{aligned}\frac{\partial^2 h_{\vec{n}_j}}{\partial\alpha^2}(\alpha) &\leqslant \sum_{\ell=0}^{n_j-1}\frac{1}{(\ell+\alpha)^2} - \frac{2}{\alpha^2}\\ &= -\frac{1}{\alpha^2} + \frac{1}{(1+\alpha)^2} + \sum_{\ell=2}^{n_j-1}\frac{1}{(\ell+\alpha)^2}\\ &< -\frac{1}{\alpha^2} + \frac{1}{(1+\alpha)^2} + \sum_{\ell=2}^{\infty}\frac{1}{\ell^2}\\ &= -\frac{1}{\alpha^2} + \frac{1}{(1+\alpha)^2} + \frac{\pi^2}{6} - 1\,,\end{aligned}$$

which is negative for any $\alpha \leqslant 1$ (the gap between the two fractions containing α obviously decreases with the increase of α, so it is enough to check the sign for the largest value $\alpha = 1$). Thus we have $\frac{\partial^2 h_{\vec{n}_j}}{\partial\alpha^2}(\alpha) < 0$. □

The concavity of $h_{\vec{n}_j}$ is useful for the following reason.

Lemma 9. *Let $S \subseteq V^{\setminus i}$ and $j \in \mathcal{D}_u^{V^{\setminus i}}$ such that $\nexists k : n_{j,k} = n_j$. If $\alpha \leqslant q(S)$ and $\frac{\partial h_{\vec{n}_j}}{\partial\alpha}(\alpha/q(S))$ is non-negative, then*

$$h_{\vec{n}_j}(\alpha/q(T)) \leqslant h_{\vec{n}_j}(\alpha/q(S)) \text{ for every } T \supseteq S.$$

Proof. Since $\nexists k : n_{j,k} = n_j$ and $\alpha/q(S) \leqslant 1$, we have that $h_{\vec{n}_j}$ is concave (Lemma 8) and since $\frac{\partial h_{\vec{n}_j}}{\partial\alpha}(\alpha/q(S)) \geqslant 0$, $h_{\vec{n}_j}$ is non-decreasing. □

The final step to improve the upper bound is to consider any local score of a parent set S as a function of the (log-)probabilities over full mass functions.

Lemma 10. *Let $S \subseteq V^{\setminus i}$ and $j_S \in \mathcal{D}_u^S$. Then*

$$\text{LLBDeu}(S, j_S) \leqslant \sum_{j \in \mathcal{D}_u^{V^{\setminus i}}:\ j \neq j^\star} ML(\vec{n}_j) + \log \Pr(\mathcal{D}_{\vec{n}_{j^\star}} | \vec{\alpha}_{j_S}),$$

where $j^\star = \arg\min_{j \in \mathcal{D}_u^{V^{\setminus i}}} \log \Pr(\mathcal{D}_{\vec{n}_j} | \vec{\alpha}_{j_S})$.

Proof. We rewrite $n_{j_S,k}$ as the sum of counts from full mass functions: $n_{j_S,k} = \sum_{j \in \mathcal{D}_u^{V^{\setminus i}}:\ j^S = j_S} n_{j,k}$. Thus, $\text{LLBDeu}(S, j_S)$ is the log probability $\log \Pr(\mathcal{D}_{\vec{n}_{j_S}} | \vec{\alpha}_{j_S})$ of observing a data sequence with counts $\vec{n}_{j_S} = (\sum_{j \in \mathcal{D}_u^{V^{\setminus i}}:\ j^S = j_S} n_{j,k})_{k \in c(i)}$ under the Dirichlet-multinomial with parameter vector $\vec{\alpha}_{j_S}$. Assume an arbitrary order for the full mass functions related to elements in $\{j \in \mathcal{D}_u^{V^{\setminus i}} : j^S = j_S\}$ and name them $j_1, \ldots, j_w$, with $w = |\{j \in \mathcal{D}_u^{V^{\setminus i}} : j^S = j_S\}|$. Exploiting the conjugacy multinomial-Dirichlet we can express this probability as a product of conditional probabilities:

$$\Pr(\mathcal{D}_{\vec{n}_{j_S}} | \vec{\alpha}_{j_S}) = \prod_{\ell=1}^{w} \Pr\left(\mathcal{D}_{\vec{n}_{j_\ell}} \middle| \sum_{t=1}^{\ell-1} \vec{n}_{j_t} + \vec{\alpha}_{j_S}\right),$$

$$\begin{aligned}\text{LLBDeu}(S, j_S) &= \sum_{\ell=1}^{w} \log \Pr\left(\mathcal{D}_{\vec{n}_{j_\ell}} \middle| \sum_{t=1}^{\ell-1} \vec{n}_{j_t} + \vec{\alpha}_{j_S}\right) \\ &\leqslant \log \Pr(\vec{n}_{j_1} | \vec{\alpha}_{j_S}) + \sum_{t=2}^{w} \text{ML}(\vec{n}_{j_t}).\end{aligned}$$

These are obtained by applying Expression (4) to all but the first term. Since the order is arbitrary, we can pick one in our best interest and the result follows. □

While the bound of Lemma 10 is valid for S, it gives no assurances about its supersets T, so it is of little direct use (if we need to compute it for every $T \supset S$, then it is better to compute the scores themselves). To address that, we replace the first term of the right-hand side summation with a proper upper bound, while the maximum likelihood terms are already valid terms, as discussed earlier. We note that Theorem 3 is in fact much simpler than its formal enunciation—this is unavoidable, since we are combining different possible bounds for the term $\log \Pr(\mathcal{D}_{\vec{n}_{j^\star}} | \vec{\alpha}_{j_S})$ that appears in Lemma 10 into one bound, while also keeping all the other maximum likelihood bounds. Moreover, to make Theorem 3 slightly more compact, we sum all maximum likelihood (ML) terms (first summation in the expression) and then we discard one of them (the first negative ML term) in order to (potentially) replace it with a better bound. This is the only reason why the definition of $\underline{h}$ in the following theorem looks unpleasant to the eyes.

Theorem 3 (ub_h)**.** *Let $S \subseteq V^{\setminus i}$, $\alpha = \alpha_{ess}/q(S)$, $j_S \in \mathcal{D}_u^S$, and $\overline{h}_{\vec{n}_j}(\alpha) = h_{\vec{n}_j}(\alpha)$ if $\alpha \leqslant 1$ and $\frac{\partial h_{\vec{n}_j}}{\partial \alpha}(\alpha) \geqslant 0$, and zero otherwise. Let*

$$\begin{aligned}\underline{h}(S, j_S) = \sum_{\substack{j \in \mathcal{D}_u^{V^{\setminus i}}: \\ j^S = j_S}} ML(\vec{n}_j) + \min_{\substack{j \in \mathcal{D}_u^{V^{\setminus i}}: \\ j^S = j_S}} \Big(&- ML(\vec{n}_j) \\ &+ \min\{ML(\vec{n}_j);\ f(V^{\setminus i}, j) + g(V^{\setminus i}, j, \alpha);\ \overline{h}_{\vec{n}_j}(\alpha)\}\Big).\end{aligned}$$

Then $\text{LLBDeu}(S, j_S) \leqslant \underline{h}(S, j_S)$. *Moreover, if* $\text{LBDeu}(S') \geqslant \sum_{j_S \in \mathcal{D}_u^S} \underline{h}(S, j_S) = \underline{h}(S)$ *for some $S' \subset S$, then S and all its supersets are not in $\mathcal{L}_i$.*

Proof. For parent set S, the bound based on $\text{ML}(\vec{n}_j)$ only (first option in the inner minimisation, which cancels out the double ML terms) is valid by Corollary 2. The other two options rely on Lemma 10 and their own results: the bound on $f(V^{\setminus i}, j) + g(V^{\setminus i}, j, \alpha)$ is valid by Lemma 6, while the bound based on $\overline{h}_{\vec{n}_j}(\alpha)$ comes from Lemma 9, and thus the result holds for S. Take $T \supset S$. It is straightforward that

$$\text{LBDeu}(T) \leqslant \sum_{j_T \in \mathcal{D}_u^T} \underline{h}(T, j_T) =$$

$$\sum_{j_S \in \mathcal{D}_u^S} \left(\sum_{j_T \in \mathcal{D}_u^T:\ j_T^S = j_S} \underline{h}(T, j_T) \right) \leqslant \sum_{j_S \in \mathcal{D}_u^S} \underline{h}(S, j_S),$$

since $\sum_{j_T \in \mathcal{D}_u^T:\ j_T^S = j_S} \underline{h}(T, j_T) \leqslant \underline{h}(S, j_S)$, because both sides run over the same full instantiations and the right-hand side use the tighter minimisation of Expression (3) only once, while the left-hand side can use that tighter minimisation once every j_T, and Lemmas 6 and 9 ensure that the computed values $f(V^{\setminus i}, j) + g(V^{\setminus i}, j, \alpha)$ and $\overline{h}_{\vec{n}_j}(\alpha)$ are valid for T. □

As with previous theorems, Theorem 3 gives us a new upper bound on the local score of a parent set S

$$\text{ub}_h(S) = \underline{h}(S) = \sum_{j_S \in \mathcal{D}_u^S} \underline{h}(S, j_S) \geqslant \max_{T : T \supset S} \text{LBDeu}(T).$$

6 Combining the Bounds

We note that bound ub_g of the previous section was obtained in a similar way as ub_f, and we prove that $\text{ub}_g(S) \leqslant \text{ub}_f(S)$ for any candidate parent set S. Conversely, ub_h bears no such relation to ub_f as we derived it through a new route, studying the properties of the likelihood function. This is to our advantage, as due to their independent theoretical derivations, ub_g and ub_h prune different regions of the search space and can be effectively combined into a tighter bound $\text{ub}_{g,h} = \min\{\text{ub}_g; \text{ub}_h\}$.

This work focus on new theoretical derivations leading to tighter bounds, and thus an empirical analysis is beyond its scope. Nonetheless, we illustrate possible gains as well as a comparison of the different bounds in simple benchmark datasets in Figures 2 and 3. The code for computing these bounds and reproducing the experiments is available on the authors' pages.

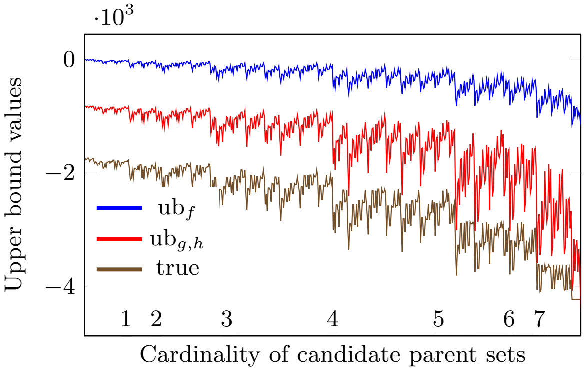

Figure 2: Upper bound values for each candidate parent set for variable Standard-of-living-index in the *CMC* dataset (Dua and Graff, 2017). Parent sets are arbitrarily ordered within each cardinality (neighbourhood in the graph within same cardinality is not relevant).

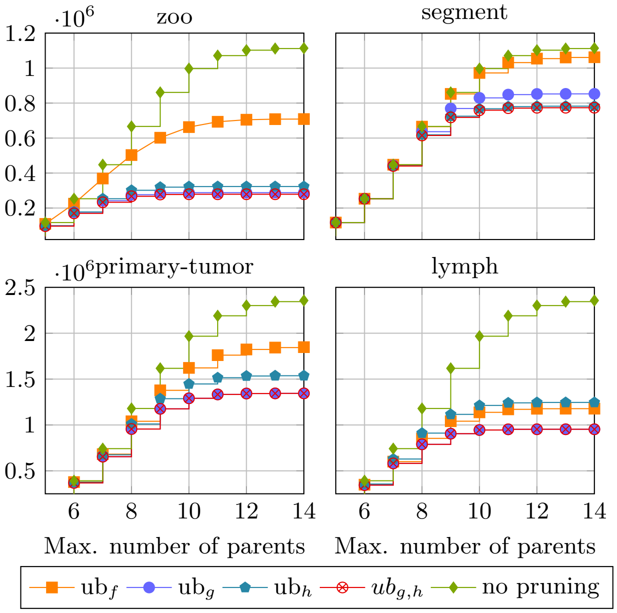

Figure 3: Number of scores computed per maximum number of parents with different pruning bounds for four UCI datasets (Dua and Graff, 2017) with 17 (zoo, segment) and 18 (primary-tumor, lymph) variables. The scores were computed using breadth-first search.

For small datasets, it is feasible to score every candidate parent set so that we can compare how far the upper bounds for a given parent set S (and all its supersets) are from the true best score among itself and its supersets. Figure 2 shows such a comparison for variable *Standard-of-living-index* in the *CMC* dataset (Dua and Graff, 2017), which has 10 variables and 1,473 instances. It is clear that the new bound $\text{ub}_{g,h}$ is much tighter than the current best bound in the literature (here called ub_f) and improves considerably towards the true best score (only available because this particular dataset is not too large).

For larger datasets (more than 10 variables), evaluating all candidate parent sets becomes computationally impracticable, so instead we evaluate the number of scores computed with each bound. In Figure 3, we see the new bounds considerably reduced the number of scores computed, which translates into smaller lists of potentially optimal parent sets L_i (see Definition 1). This goes to show the practical value of tighter upper bounds, as we save computing time in both steps of BNSL: parent set identification (fewer scores to compute) and structure optimisation (smaller search space).

Finally, we point out that the mathematical results may seem harder to apply than they actually are. Computing $\text{ub}_g(S)$ and $\text{ub}_h(S)$ to prune a parent set S and all its supersets can be done in linear time, as one pass through the data is enough to collect and process all required counts; more sophisticated data structures, such as AD-trees (Moore and Lee, 1998), might allow for even greater speedups. Since calculating a score already takes linear time in the number of data samples, we have cheap bounds which are provably superior to the current state-of-the-art pruning for BDeu.

7 Conclusions

We introduced new theoretical upper bounds for exact structure learning of Bayesian networks with the BDeu score by studying the score function from multiple angles. These bounds are provably tighter than previous results and shall provide significant benefits in reducing the search space in candidate parent set identification in BNSL and potentially other applications involving independence assumptions.

A natural step for future research is the integration of our bounds with more sophisticated data structures and search algorithms. As an example, branch-and-bound methods are particularly promising as they not only consider the parent sets and its corresponding full instantiations but also partial instantiations that are formed by disallowing some variables to be parents in some of the branches. Our results also open new routes for further theoretical work in exact structure learning. Notably, we conjecture that the maximum likelihood estimation terms still leave room for tighter bounds.

References

Bartlett, M. and Cussens, J. (2017). Integer linear programming for the Bayesian network structure learning problem. *Artificial Intelligence*, 244:258–271.

Chen, E. Y.-J., Shen, Y., Choi, A., and Darwiche, A. (2016). Learning Bayesian networks with ancestral constraints. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29 (NIPS)*, pages 2325–2333. Curran Associates, Inc.

Chickering, D. M., Heckerman, D., and Meek, C. (2004). Large-sample learning of Bayesian networks is NP-hard. *Journal of Machine Learning Research*, 20:1287–1330.

Cooper, G. F. and Herskovits, E. (1992). A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning*, 9:309–347.

Cussens, J. (2011). Bayesian network learning with cutting planes. In *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 153–160. AUAI Press.

Cussens, J. (2012). An upper bound for BDeu local scores. Proc. ECAI-2012 workshop on algorithmic issues for inference in graphical models (AIGM).

Cussens, J. and Bartlett, M. (2015). GOBNILP 1.6.2 User/Developer Manual.

Cussens, J., Bartlett, M., Jones, E. M., and Sheehan, N. A. (2013). Maximum likelihood pedigree reconstruction using integer linear programming. *Genetic Epidemiology*, 37(1):69–83.

de Campos, C. and Ji, Q. (2010). Properties of Bayesian Dirichlet scores to learn Bayesian network structures. In *Conference on Advancements in Artificial Intelligence (AAAI)*, pages 431–436.

de Campos, C. and Ji, Q. (2011). Efficient structure learning of Bayesian networks using constraints. *Journal of Machine Learning Research*, 12:663–689.

de Campos, C., Zeng, Z., and Ji, Q. (2009). Structure learning of Bayesian networks using constraints. In *Proc. of the 26th International Conference on Machine Learning (ICML)*, volume 382, pages 113–120. ACM.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

Heckerman, D., Geiger, D., and Chickering, D. M. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243.

Jaakkola, T., Sontag, D., Globerson, A., and Meila, M. (2010). Learning Bayesian network structure using LP relaxations. In *Proceedings of 13th International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, volume 9, pages 358–365. Journal of Machine Learning Research Workshop and Conference Proceedings.

Koivisto, M. (2006). Parent Assignment Is Hard for the MDL, AIC, and NML Costs. In *Computational Learning Theory (COLT)*, volume 4005, pages 289–303. Springer.

Koivisto, M. and Sood, K. (2004). Exact Bayesian structure discovery in Bayesian networks. *Journal of Machine Learning Research*, 5:549–573.

Levin, B. and Reeds, J. (1977). Compound multinomial likelihood functions are unimodal: Proof of a conjecture of I. J. Good. *The Annals of Statistics*, 5(1):79–87.

Moore, A. and Lee, M. S. (1998). Cached sufficient statistics for efficient machine learning with large datasets. *Journal of Artificial Intelligence Research*, 8:67–91.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.

Teyssier, M. and Koller, D. (2005). Ordering-based search: A simple and effective algorithm for learning Bayesian networks. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 584–590.

Yuan, C. and Malone, B. (2012). An improved admissible heuristic for learning optimal Bayesian networks. In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 924–933, Catalina Island, CA.

Yuan, C. and Malone, B. (2013). Learning optimal Bayesian networks: A shortest path perspective. *Journal of Artificial Intelligence Research*, 48:23–65.