Collective Graphical Models Supplementary Material

Daniel Sheldon
Oregon State University
sheldon@eecs.oregonstate.edu

Thomas G. Dietterich
Oregon State University
tgd@eecs.oregonstate.edu

1 Initialization

The overall initialization is a sequence of three algorithms. The first algorithm joins two tables $\mathbf{n}_A$ and $\mathbf{n}_B$ to find a table $\mathbf{n}_{A\cup B}$ such that $\mathbf{n}_A, \mathbf{n}_B \preceq \mathbf{n}_{A\cup B}$. The second algorithm extends a locally consistent configuration $\mathbf{n}_{\mathcal{A}}$ to find a single table $\mathbf{n}_V \in \text{tbl}(V)$ such that $\mathbf{n}_{\mathcal{A}} \preceq \mathbf{n}_V$, thus providing a constructive proof of Theorem 1. The third algorithm finds a configuration $\mathbf{n}_{\mathcal{C}}$ such that $\mathbf{n}_{\mathcal{D}} \preceq \mathbf{n}_{\mathcal{C}}$ whenever: (i) $\mathcal{D}$ is decomposable, (ii) $\mathbf{n}_{\mathcal{D}}$ is consistent, and (iii) $\mathcal{D} \preceq C$; thus solving the initialization problem.

1.1 Joining two tables

Let $\mathbf{n}_A$ and $\mathbf{n}_B$ be two tables having a common marginal table $\mathbf{n}_{A\cap B} \preceq \mathbf{n}_A, \mathbf{n}_B$. The join operation, denoted $\mathbf{n}_A \vee \mathbf{n}_B$, constructs a table $\mathbf{n}_{A\cup B} := \mathbf{n}_A \vee \mathbf{n}_B \in \text{tbl}(A \cup B)$ that extends both $\mathbf{n}_A$ and $\mathbf{n}_B$, meaning that $\mathbf{n}_A, \mathbf{n}_B \preceq \mathbf{n}_{A\cup B}$. The key observation is that for each $j \in \mathcal{X}_{A\cap B}$, the subtable $\mathbf{n}_{A\cup B}(\cdot, j, \cdot)$ can be viewed as a two-dimensional table with row sums that are determined by $\mathbf{n}_A$, column sums that are determined by $\mathbf{n}_B$, and grand total $n_{A\cap B}(j)$. Let $A' = A \setminus B,\ B' = B \setminus A$. Then $\mathbf{n}_{A\cup B}$ must satisfy

$$n_A(i,j) = \sum_{k\in\mathcal{X}_{B'}} n_{A\cup B}(i,j,k),\ \ \forall i \in \mathcal{X}_{A'}, \quad n_B(j,k) = \sum_{i\in\mathcal{X}_{A'}} n_{A\cup B}(i,j,k),\ \ \forall k \in \mathcal{X}_{B'}.$$

By our assumption that $\mathbf{n}_{A\cap B} \preceq \mathbf{n}_A, \mathbf{n}_B$, the vectors $\mathbf{r} = (n_A(i,j))_{i\in\mathcal{X}_{A'}}$ and $\mathbf{c} = (n_B(j,k))_{k\in\mathcal{X}_{B'}}$ share the common grand total $n_{A\cap B}(j)$. The problem of finding a two-dimensional table $\mathbf{n}_{A\cup B}(\cdot, j, \cdot)$ with rows sums $\mathbf{r}$ and column sums $\mathbf{c}$ is equivalent to finding a feasible solution to the *transportation problem* [1]. Any variant of the following method is correct: start with an all-zero table and repeatedly (1) select a row and column whose current sums are smaller than the specified sums, (2) add an integer amount to the entry in that row and column without exceeding the specified row and column sums. This process is repeated for each $j \in \mathcal{X}_{A\cap B}$ to provide an algorithm for the join operation.

1.2 Constructing $\mathbf{n}_V$ by edge contraction in $\mathcal{T}_{\mathcal{A}}$

The join operation can be used to construct a complete contingency table $\mathbf{n}_V$ by a sequence of simple operations on the junction tree. Let $\mathbf{n}_{\mathcal{A}}$ be a locally consistent configuration on the junction tree $\mathcal{T}_{\mathcal{A}}$. Define the *contraction* of edge $(A, B) \in \mathcal{E}(\mathcal{T}_{\mathcal{A}})$ to be the following operation, which simultaneously updates the collection $\mathcal{A}$, junction tree $\mathcal{T}_{\mathcal{A}}$, and tables $\mathbf{n}_{\mathcal{A}}$, which we collect in the tuple $(\mathcal{A}, \mathcal{T}_{\mathcal{A}}, \mathbf{n}_{\mathcal{A}})$. First, update $\mathcal{A}$ by replacing the sets A and B by their union $A \cup B$; next, update $\mathcal{T}_{\mathcal{A}}$ by connecting $A \cup B$ to all former neighbors of A and B; finally, update $\mathbf{n}_{\mathcal{A}}$ by replacing $\mathbf{n}_A$ and $\mathbf{n}_B$ by $\mathbf{n}_{A\cup B} = \mathbf{n}_A \vee \mathbf{n}_B$. Local consistency ensures that $\mathbf{n}_A$ and $\mathbf{n}_B$ agree on $A \cap B$ and hence the join operation is possible. The overall algorithm for constructing $\mathbf{n}_V$ is then quite simple: repeatedly contract edges until $\mathcal{A}$ consists of the single set V, at which point the remaining table $\mathbf{n}_V \in \text{tbl}(V)$ satisfies $\mathbf{n}_A \preceq \mathbf{n}_V$ for all $A \in \mathcal{A}$.

Proof of correctness (Theorem 1). To prove the correctness of this procedure and thus establish Theorem 1, we must argue that the following properties hold for the tuple $(\mathcal{A}', \mathcal{T}_{\mathcal{A}'}, \mathbf{n}_{\mathcal{A}'})$ that is the result of the contraction: (1) $\mathcal{T}_{\mathcal{A}'}$ is a junction tree, (2) $\mathbf{n}_{\mathcal{A}'}$ is locally consistent and (3) $\mathbf{n}_{\mathcal{A}} \preceq \mathbf{n}_{\mathcal{A}'}$.

To show that $\mathcal{T}_{\mathcal{A}'}$ is a junction tree we use the characterization of junction trees that $\mathcal{T}_{\mathcal{A}'}$ is a junction tree if for all $v \in V$, the subgraph $\mathcal{T}'_v$ induced by $\{A \in \mathcal{A}' : v \in A\}$ is connected. Define $\mathcal{T}_v$ analogously based on $\mathcal{T}_{\mathcal{A}}$. It is straightforward to see that $\mathcal{T}'_v$ is obtained from $\mathcal{T}_v$ by one of the following operations: (1) if A and B are both present in $\mathcal{T}_v$, then contract (A, B) into the single clique $A \cup B$ to obtain $\mathcal{T}'_v$, (2) if exactly one of A or B is present in $\mathcal{T}_v$, then replace that clique by $A \cup B$ to obtain $\mathcal{T}'_v$, (3) otherwise, make no change to $\mathcal{T}_v$. In each case, the connectedness of $\mathcal{T}'_v$ follows directly from that of $\mathcal{T}_v$.

To see that $\mathbf{n}_{\mathcal{A}'}$ is locally consistent, let $(A \cup B, C) \in \mathcal{E}(T_{\mathcal{A}'})$ be an edge involving the newly created set $A \cup B$, which means that either A or B was a neighbor of C in $\mathcal{T}_{\mathcal{A}}$; assume without loss of generality that $(B, C) \in \mathcal{E}(\mathcal{T}_{\mathcal{A}})$. Thus B is on the path from A to C in $\mathcal{T}_{\mathcal{A}}$, and the running intersection property implies that $A \cap C \subseteq B$, which in turn implies that $(A \cup B) \cap C = B \cap C$. Thus the separator for this edge is $B \cap C$, and the consistency requirement is that $\mathbf{n}_{A\cup B} \downarrow B \cap C = \mathbf{n}_C \downarrow B \cap C$.

Starting with $\mathbf{n}_{A\cup B}$, we have that

$$\mathbf{n}_{A\cup B} \downarrow B \cap C = (\mathbf{n}_{A\cup B} \downarrow B) \downarrow B \cap C = \mathbf{n}_B \downarrow B \cap C.$$

In the the first equality, we compute $\mathbf{n}_{A\cup B} \downarrow B \cap C$ by first marginalizing onto B and then onto the subset $B \cap C$, which does not change the final result. In the second equality we use the fact that the $\mathbf{n}_B \preceq \mathbf{n}_{A\cup B}$ by construction in the join operation, and hence $\mathbf{n}_{A\cup B} \downarrow B = \mathbf{n}_B$.

Starting with $\mathbf{n}_C$ we have

$$\mathbf{n}_C \downarrow B \cap C = \mathbf{n}_B \downarrow B \cap C$$

by local consistency of the tables on $\mathcal{T}_{\mathcal{A}}$. Since these expressions are equal, we have established local consistency for the edge from $(A \cup B, C)$ where C was chosen arbitrarily; hence, local consistency holds for all edges involving the newly created set $A \cup B$. For all other edges, the tables remain unchanged and hence the consistency condition continues to hold from $\mathcal{T}_{\mathcal{A}}$.

It remains only to check that $\mathbf{n}_{\mathcal{A}} \preceq \mathbf{n}_{\mathcal{A}'}$. The only difference between these configurations is that $\mathbf{n}_A$ and $\mathbf{n}_B$ in the former were replaced by $\mathbf{n}_{A\cup B} = \mathbf{n}_A \vee \mathbf{n}_B$ in the latter, for which $\mathbf{n}_A, \mathbf{n}_B \preceq \mathbf{n}_{A\cup B}$ by construction.

Because one edge is removed in each contraction while preserving the ground set V, the overall contraction procedure will terminate in $|\mathcal{E}(\mathcal{T})|$ steps with $\mathcal{A} = \{V\}$. The relation $\preceq$ is clearly transitive, and hence we have established that $\mathbf{n}_A \preceq \mathbf{n}_V$ for the final table $\mathbf{n}_V$, which is our desired result. □

As a side comment, we note that it is slightly less work to prove Theorem 1 using a divide-and-conquer scheme, which can be seen to be equivalent to a sequence of edge contractions, each of which joins a leaf of $\mathcal{T}_{\mathcal{A}}$ with its unique neighbor. However, the flexibility of scheduling contractions in any order is essential for the following algorithm.

1.3 Constructing $\mathbf{n}_{\mathcal{C}}$ from $\mathbf{n}_{\mathcal{D}}$

Recall that our goal for initializing the Markov chain is to populate a configuration $\mathbf{n}_{\mathcal{C}}$ such that $\mathbf{n}_{\mathcal{D}} \preceq \mathbf{n}_{\mathcal{C}}$ when $\mathcal{D} \preceq \mathcal{C}$. We may assume that $\bigcup \mathcal{D} = V$ (if not, construct initial tables $\mathbf{n}_v$ arbitrarily for each $v \in U := (V \setminus \bigcup \mathcal{D})$, and the collection $\mathcal{D}$ augmented by the singletons $v \in U$ remains decomposable). Thus, an initialization approach that is correct but computationally infeasible is to first build the full table $\mathbf{n}_V$ by contracting all edges of $\mathcal{T}_{\mathcal{D}}$ and then form the marginal tables $\mathbf{n}_C = \mathbf{n}_V \downarrow C$. Instead, this procedure can be modified to intersperse edge contractions with marginalization steps.

The operations are sequenced in a collect phase and a distribute phase on the junction tree $\mathcal{T}_{\mathcal{C}}$ for collection $\mathcal{C}$, with an arbitrarily chosen root node R. The algorithm maintains the tuple $(\mathcal{A}, \mathcal{T}_{\mathcal{A}}, \mathbf{n}_{\mathcal{A}}, \pi)$, where the final entry $\pi : \mathcal{A} \to \mathcal{C}$ is a function such that $A \subseteq \pi(A)$ that is a witness to the relation $\mathcal{A} \preceq \mathcal{C}$. When $C = \pi(A)$, we refer to C as the *owner* of A. Initially, $(\mathcal{A}, \mathcal{T}_{\mathcal{A}}, \mathbf{n}_{\mathcal{A}}) = (\mathcal{D}, \mathcal{T}_{\mathcal{D}}, \mathbf{n}_{\mathcal{D}})$, and $\pi : \mathcal{D} \to \mathcal{C}$ is chosen to assign each $D \in \mathcal{D}$ an owner in $\mathcal{C}$.

The operation COLLECT(C) has the effect of conducting the following operations on the subtree of $\mathcal{T}_\mathcal{C}$ rooted at C: first, contract all edges of $\mathcal{T}_D$ with both endpoints owned by the subtree (i.e. owned by C or one of its descendants); then marginalize each remaining table in the subtree onto C. Following the construction of a complete table $\mathbf{n}_C$, the operation DISTRIBUTE(C) then completes the tables for all descendants of C. Detailed descriptions of COLLECT and DISTRIBUTE are given in Table 1.

The overall algorithm is to call COLLECT(R), which contracts all edges and terminates with $\mathcal{A} = \{R\}$ and $\mathbf{n}_\mathcal{A} = \{\mathbf{n}_R\}$. The complete table $\mathbf{n}_R$ for the root node is then extracted, and DISTRIBUTE(R) is called to complete the remaining tables.

INITIALIZE

1. Pick an arbitrary root clique $R \in \mathcal{C}$
2. Execute COLLECT(R)
3. Execute DISTRIBUTE(R)

COLLECT(C)

1. For each child C' of C, do the following:
   (a) Call COLLECT(C')
   (b) Marginalize out the variables in $C' \setminus C$ from each set $A \in \mathcal{A}$:
      i. Update $\mathcal{A}$ to replace A by $A \cap (C \setminus C')$
      ii. Update $\mathbf{n}_\mathcal{A}$ to replace $\mathbf{n}_A$ by $\mathbf{n}_{A\cap(C\setminus C')} := \mathbf{n}_A \downarrow A \cap (C \setminus C')$
   (c) Update π to transfer ownership of all sets from child to parent: if A was owned by C', then set $\pi(A \cap (C \setminus C')) = C$
2. Repeatedly contract edges $(A, B) \in \mathcal{E}(\mathcal{T}_\mathcal{A})$ with $\pi(A) = \pi(B) = C$ (both endpoints are owned by C) and set $\pi(A \cup B) = C$ until no additional contractions are possible.
3. Let $\mathcal{A}_C = \pi^{-1}(C)$ be the members of $\mathcal{A}$ owned by C *after* contraction. Save the corresponding tables $\mathbf{n}_{\mathcal{A}_C}$ for use in the distribute phase.

DISTRIBUTE(C)

1. Assume that the table $\mathbf{n}_C$ has been constructed
2. For each child C' of C, do the following:
   (a) Let $S = C \cap C'$, and let $\mathbf{n}_S = \mathbf{n}_C \downarrow S$
   (b) Suppose that $\mathbf{n}_{\mathcal{A}_{C'}} = \{\mathbf{n}_{A_1}, \ldots, \mathbf{n}_{A_\ell}\}$
   (c) Let $\mathbf{n}_{C'} = \mathbf{n}_S \vee \mathbf{n}_{A_1} \vee \ldots \vee \mathbf{n}_{A_\ell}$ (joins may be done in any order)

Table 1: Initialization

Theorem S.1. *The algorithm* INITIALIZE *in Table 1 terminates with a consistent configuration* $\mathbf{n}_\mathcal{C}$ *such that* $\mathbf{n}_\mathcal{D} \preceq \mathbf{n}_\mathcal{C}$.

Proof. The proof of correctness must first argue that the tables involved in each join operation are consistent. To do this, we show that during the collect phase, each operation preserves the invariant that $\mathcal{T}_\mathcal{A}$ is a junction tree and $\mathbf{n}_\mathcal{A}$ is consistent. We already showed in the proof of Theorem 1 that edge contractions preserve these properties. The only other modifications are made by the marginalization operations in Step 1(b) of COLLECT, which remove the variables in $(C \setminus C')$ from every set in $T_\mathcal{A}$ and each table in $\mathbf{n}_\mathcal{A}$ (by marginalization). These operations clearly preserve the running intersection property of junction trees, as well as local consistency.

In the distribute phase, we will show that the configuration $\{\mathbf{n}_S, \mathbf{n}_{A_1}, \ldots, \mathbf{n}_{A_\ell}\}$ is locally consistent on the "star" junction tree $\mathcal{T}^*$ where S is connected to each other set, and hence the joins may be viewed as edge contractions. To see that $\mathcal{T}^*$ is indeed a junction tree, let $A_1, A_1 \in \mathcal{A}_{C'}$ and let $(\mathcal{A}, \mathcal{T}_\mathcal{A}, \mathbf{n}_\mathcal{A}, \pi)$ be the state variables from the point in time immediately following the execution COLLECT(C'). Then it must be the case that $A_1, A_2 \in \mathcal{A}$ and the path from A_1 to A_2 in $\mathcal{T}_\mathcal{A}$ contains some set A_3 for which $\pi(A_3) \neq C'$; otherwise the entire path would have been contracted. Thus, from the running intersection property of $\mathcal{T}_{\mathcal{A}}$, we have that

$$A_1 \cap A_2 \subseteq A_3.$$

Furthermore, $\pi(A_3)$ is not a descendant of C', because all sets owned by descendants have been transferred to C'. Thus the unique path from C' to $\pi(A_3)$ in $\mathcal{T}_{\mathcal{C}}$ must go through the parent C, implying by the running intersection property of $\mathcal{T}_{\mathcal{C}}$ that

$$C' \cap \pi(A_3) \subseteq S.$$

Finally, we have by the definition of ownership that $A_1 \subseteq C'$ and $A_3 \subseteq \pi(A_3)$, so we may write the following chain of equalities and inclusions:

$$\begin{aligned} A_1 \cap A_2 &= A_1 \cap (A_1 \cap A_2) \\ &\subseteq A_1 \cap A_3 \\ &\subseteq \pi(A_1) \cap \pi(A_3) \\ &= C' \cap \pi(A_3) \\ &\subseteq S. \end{aligned}$$

This establishes the running intersection property on $\mathcal{T}^*$.

To see that the configuration $\{\mathbf{n}_S, \mathbf{n}_{A_1}, \ldots, \mathbf{n}_{A_\ell}\}$ is locally consistent on $\mathcal{T}^*$, we note that $A_i \cap S = A_i \cap (C \setminus C')$ because $A_i \subseteq C'$. By construction in $\textsc{collect}(C')$, we have that

$$\mathbf{n}_{A_i \cap S} = \mathbf{n}_{A_i \cap (C \setminus C')} \preceq \mathbf{n}_{A_i}.$$

Then, by construction in $\textsc{distribute}(C)$ (for the parent), we have that

$$\mathbf{n}_{A_i \cap S} = \mathbf{n}_{A_i \cap (C \setminus C')} \preceq \mathbf{n}_S.$$

This establishes consistency for each join operation executed by COLLECT and DISTRIBUTE.

To verify the final consistency of $\mathbf{n}_{\mathcal{C}}$, it is easy to see in the distribute phase that $\mathbf{n}_S \preceq \mathbf{n}_C, \mathbf{n}_{C'}$ and hence the configuration $\mathbf{n}_{\mathcal{C}}$ is locally consistent by construction, and thus globally consistent by Theorem 1. Finally, to check that $\mathbf{n}_{\mathcal{D}} \preceq \mathbf{n}_{\mathcal{C}}$, suppose that $\pi(D) = C$. Then, after $\textsc{collect}(C)$, there is some D' obtained from one or more join operations involving D such that $D \subseteq D' \in \mathcal{A}_C$, and hence $\mathbf{n}_D \preceq \mathbf{n}_{D'}$. The execution of $\textsc{distribute}(C)$ conducts further joins on D' but guarantees that $\mathbf{n}_{D'} \preceq \mathbf{n}_C$, so that $\mathbf{n}_D \preceq \mathbf{n}_C$. This proves the result. □

2 Markov basis (Theorem 3)

Proof of Theorem 3. Let $\mathbf{n}, \mathbf{n}' \in \mathcal{F}^*_{\mathbf{n}_{\mathcal{D}}}$. We will prove the special case when $\mathcal{U} = \{U\}$. Let $\{\mathbf{x}^{(m)} : m = 1, \ldots, M\}$ be an arbitrarily ordered sample corresponding to contingency table $\mathbf{n}$ and define $\mathbf{x}'^{(m)}$ analogously for the table $\mathbf{n}'$. Define $\mathbf{z}^{(m)} \in \mathcal{M}^{d=1}(U, W)$ to have the nonzero entries $z(\mathbf{x}_U^{(m)}, \mathbf{x}_W^{(m)}) = -1$ and $z(\mathbf{x}_U'^{(m)}, \mathbf{x}_W^{(m)}) = 1$; this move updates $\mathbf{x}^{(m)}$ to match $\mathbf{x}'^{(m)}$ on the variables in U. The moves may be executed in any order and maintain a valid sample and hence a non-negative table. The table also remains in $\mathcal{F}^*_{\mathbf{n}_{\mathcal{D}}}$ because the entire $\mathbf{n}_W$ marginal is preserved, and the $\mathbf{n}_U$ marginal is unrestricted. Define $\mathbf{n}'' = \mathbf{n} + \sum_{m=1}^M \mathbf{z}^{(m)}$. By construction, we now have that $\mathbf{n}'' \downarrow U = \mathbf{n}' \downarrow U$, and we have maintained the property that $\mathbf{n}'' \downarrow D = \mathbf{n} \downarrow D = \mathbf{n}' \downarrow D$ for all $D \in \mathcal{D}$.

Since $\mathcal{D}$ and $\mathcal{U}$ are decomposable on disjoint ground sets, $\mathcal{D}' = \mathcal{D} \cup \mathcal{U}$ is decomposable. By construction, $\bigcup \mathcal{D}' = V$, so the conditions are met for the Dobra basis $\mathcal{M}_{\mathcal{D}'}$. Hence there is a sequence of moves in $\mathcal{M}_{\mathcal{D}'}$ connecting $\mathbf{n}''$ to $\mathbf{n}'$, which proves the result.

The generalization to arbitrary decomposable collections $\mathcal{U}$ is straightforward by repeating the argument we just made to adjust the variables in sequence for each set $U \in \mathcal{U}$ in an order dictated by a junction tree $\mathcal{T}_{\mathcal{U}}$. □

Proposition S.1. *For any degree two move $\mathbf{z}$ generated by the partition (A, S, B), the set of cliques $C \in \mathcal{C}$ such that $\mathbf{z} \downarrow C$ is nonzero form a connected subtree of $\mathcal{T}_{\mathcal{C}}$.*

Proof. By the junction tree property, the cliques containing A induce a connected subtree of $\mathcal{T}_\mathcal{C}$, as do the cliques containing B. The intersection of two subtrees is also a tree. $\square$

Proposition S.2. *For any degree one move* $\mathbf{z}$*, the set of cliques* C *such that* $\mathbf{z} \downarrow C$ *is nonzero form a connected subtree of* $\mathcal{T}_\mathcal{C}$*.*

Proof. The fact that the cliques that intersect A form a subtree is a direct consequence of the junction tree property. $\square$

3 Log-concavity (Theorem 4)

Before proving Theorem 4, we state and prove the following lemma.

Lemma S.1. *Let* $\mathbf{z}$ *be a degree-two move from* $\mathcal{M}^{d=2}(A, S, B)$*. Then for all* $F \subseteq V$*, it is either the case that* $\mathbf{z} \downarrow F$ *is also a degree-two move, from the set* $\mathcal{M}^{d=2}(A \cap F, S \cap F, B \cap F)$*, or that* $\mathbf{z} \downarrow F = \mathbf{0}$*. Similarly, for a degree-one move* $\mathbf{z} \in \mathcal{M}^{d=1}(A, B)$*, it is either the case that* $\mathbf{z} \downarrow F$ *is a degree-one move from the set* $\mathcal{M}^{d=1}(A \cap F, B \cap F)$*, or that* $\mathbf{z} \downarrow F = 0$*.*

Proof of Lemma S.1. We can write the degree two move defined in (4) as $\mathbf{z} = \mathbf{z}^+ - \mathbf{z}^-$ where $\mathbf{z}^+$ and $\mathbf{z}^-$ each have two positive entries

$$\mathcal{I}(\mathbf{z}^+) = \{(i, j, k), (i', j, k')\}, \quad \mathcal{I}(\mathbf{z}^-) = \{(i', j, k), (i, j, k')\},$$

We then have that $\mathbf{z} \downarrow F = (\mathbf{z}^+ \downarrow F) - (\mathbf{z}^- \downarrow F)$ where

$$\mathcal{I}(\mathbf{z}^+ \downarrow F) = \{(i_F, j_F, k_F), (i'_F, j_F, k'_F)\},$$
$$\mathcal{I}(\mathbf{z}^- \downarrow F) = \{(i'_F, j_F, k_F), (i_F, j_F, k'_F)\}.$$

In this case we use the notation that $i_F \in \mathcal{X}_{A \setminus F}$ is the subvector of i corresponding to variables in $A \cap F$, and use similar notation for j_F, k_F, etc. If either $i_F = i'_F$ or $k_F = k'_F$, then $\mathcal{I}(\mathbf{z}^+ \downarrow F) = \mathcal{I}(\mathbf{z}^- \downarrow F)$ which implies that $\mathbf{z} \downarrow F = 0$. Otherwise, the four entries are unique and $\mathbf{z} \downarrow F$ has the form of a degree two move from $\mathcal{M}^{d=2}(A \cap F, S \cap F, B \cap F)$.

The proof for degree-one moves is similar. $\square$

Proof of Theorem 4. We must show that $p(\delta)^2 \geq p(\delta - 1)p(\delta + 1)$ for all $\delta \in \{\delta_{\min}, \ldots, \delta_{\max}\}$. Defining $r(\delta) = p(\delta)/p(\delta - 1)$, it is equivalent to show that $r(\delta) \geq r(\delta + 1)$. From the factorization in (6), we can write

$$r(\delta) = \prod_{C \in \mathcal{C}(\mathbf{z})} r_C(\delta) \prod_{S \in \mathcal{S}(\mathbf{z})} r_S(\delta)$$

where $r_A(\delta) = p_A(\delta)/p_A(\delta - 1)$ for $A \in \mathcal{C} \cup \mathcal{S}$.

From (7) we see that

$$\begin{aligned} r_C(\delta) &= \prod_{i \in \mathcal{I}^+(\mathbf{z}_C)} \frac{\mu_C(i)}{(n_C(i) + \delta)} \prod_{j \in \mathcal{I}^-(\mathbf{z}_C)} \frac{(n_C(j) - \delta + 1)}{\mu_C(j)} \\ &\propto \prod_{i \in \mathcal{I}^+(\mathbf{z}_C)} (n_C(i) + \delta)^{-1} \prod_{j \in \mathcal{I}^-(\mathbf{z}_C)} (n_C(j) - \delta + 1) \end{aligned} \tag{S.1}$$

where in the final expression we ignore terms that are constant with respect to δ. All terms in (S.1) are non-negative for δ in the specified range, and each is decreasing in δ, and thus $r_C(\delta) > r_C(\delta+1)$ for each C. Thus, $p_C(\delta)$ is log-concave for all C.

However, the same reasoning implies that p_S is log-concave for all S, and because these terms appear in the denominator of (6), a further argument is required to show that $p(\delta)$ is log-concave. Consider an arbitrary separator $S = C \cap C'$ with $(C, C') \in \mathcal{E}(\mathcal{T}_\mathcal{C})$. If $\mathbf{z}_S$ is nonzero, then both $\mathbf{z}_C$ and $\mathbf{z}_{C'}$ are nonzero, because $\mathbf{z}_S$ is a marginal move of each. Thus we may assign each separator

$S \in \mathcal{S}(\mathbf{z})$ to a unique clique $C \in \mathcal{C}(\mathbf{z})$ by orienting the edges of $\mathcal{T}_{\mathcal{C}}$ toward an arbitrary root clique and assigning S to its parent. We can then write

$$p(\delta) = \prod_{\ell=1}^{L} \frac{p_{C_\ell}(\delta)}{p_{S_\ell}(\delta)} \prod_{\ell=L+1}^{L'} p_{C_\ell}(\delta)$$

where $S_\ell \subseteq C_\ell$ for $\ell = 1, \ldots, L$ and the cliques C_ℓ for $\ell > L$ are those that are not assigned a separator. Since a product of log-concave distributions is log-concave, and we have already shown that each $p_{C_\ell}(\delta)$ is log-concave, it now suffices to show that $p_{C_\ell}(\delta)/p_{S_\ell}(\delta)$ is log-concave for each $\ell = 1, \ldots, L$.

To that end, fix ℓ and let $C = C_\ell$ and $S = S_\ell$. We will show that $p_C(\delta)/p_S(\delta)$ is log-concave by using Lemma S.1 to match terms of $r_C(\delta)$ and $r_S(\delta)$. Consider first the case when $\mathbf{z}$ is a degree-two move. Then both $\mathbf{z}_C$ and $\mathbf{z}_S$, which are nonzero, are also degree-two moves with two positive and two negative entries. For $\mathbf{z}_S$, write the positive indices as $\mathcal{I}^+(\mathbf{z}_S) = \{i^{11}, i^{22}\}$ and the negative indices as $\mathcal{I}^-(\mathbf{z}_S) = \{i^{12}, i^{21}\}$ to match the 2×2 visualization $\begin{smallmatrix} + & - \\ - & + \end{smallmatrix}$.

Because $z_S(i^{11}) = \sum_{j \in \mathcal{X}_{C \setminus S}} z_C(i^{11}, j) = 1$ and we know that no cancellation occurs in the sum because both $\mathbf{z}_C$ and $\mathbf{z}_S$ have four nonzero entries, there is a unique j^{11} such that $z_C(i^{11}, j^{11}) = 1$. This argument clearly extends to find the unique j^{ab} such that $z_S(i^{ab}) = z_C(i^{ab}, j^{ab})$ for $a, b \in \{1, 2\}$. Now, for shorthand, write $n_S^{ab} = n_S(i^{ab})$ and $n_C^{ab} = n_C(i^{ab}, j^{ab})$. It is clearly the case that $n_C^{ab} \leq n_S^{ab}$ because the latter is a marginal total that include the former. At this point we can rewrite (S.1) as

$$r_C(\delta) \propto \frac{(n_C^{12} - \delta + 1)(n_C^{21} - \delta + 1)}{(n_C^{11} + \delta)(n_C^{22} + \delta)} \tag{S.2}$$

Using (S.2) and the analogous derivation for $r_S(\delta)$, we obtain

$$\frac{r_C(\delta)}{r_S(\delta)} \propto \frac{(n_C^{12} - \delta + 1)}{(n_S^{12} - \delta + 1)} \cdot \frac{(n_C^{21} - \delta + 1)}{(n_S^{21} - \delta + 1)} \cdot \frac{(n_S^{11} + \delta)}{(n_C^{11} + \delta)} \cdot \frac{(n_S^{22} + \delta)}{(n_C^{22} + \delta)}$$

The first two terms are of the form $\frac{a-\delta}{b-\delta}$ for $0 \leq a \leq b$, and thus are decreasing in δ. The latter two terms have the form $\frac{b+\delta}{a+\delta}$ for $0 \leq a \leq b$, and are also decreasing in δ. Thus, $r_C(\delta)/r_S(\delta)$ is decreasing, which implies that $p_C(\delta)/p_S(\delta)$ is log-concave. This proves the result for degree-two moves. The proof is similar for degree-one moves. □

4 Experiments

4.1 Details of convergence experiments

The additional details of the convergence experiments are as follows. Each Bayes net had 10 binary variables, a random graph structure (directed, acyclic, indegree at most 3) and random parameters. To derive the CGM, a junction tree was found for each net by the standard process of moralization and triangulation.

We then ran $K = 30$ trials for each net. In the kth trial, we generated observations $\mathbf{n}_{\mathcal{D}}^k$ by sampling from the CGM distribution, and then used our sampler to produce estimates $\hat{\mathbf{n}}_{\mathcal{C}}^{k,t}$ of $E[\mathbf{n}_{\mathcal{C}} \mid \mathbf{n}_{\mathcal{D}}^k]$ as a function of the number of MCMC steps t. We wish to explore the convergence of the estimate $\hat{\mathbf{n}}_{\mathcal{C}}^{k,t}$ with respect to t, but do not know another algorithm to compute the correct answer for comparison.

To get around this, we use the fact that $E[E[\mathbf{n}_{\mathcal{C}} \mid \mathbf{n}_{\mathcal{D}}]] = E[\mathbf{n}_{\mathcal{C}}]$ (a basic property of conditional expectation) which implies that, if our estimates of the conditional expectation are correct, then averaging over enough trials will give us back the unconditional expectation $E[\mathbf{n}_{\mathcal{C}}] = M\boldsymbol{\mu}_C$, which we know from the model parameters. Specifically, $\bar{\mathbf{n}}_{\mathcal{C}}^t := K^{-1} \sum_{k=1}^{K} \hat{\mathbf{n}}_{\mathcal{C}}^{k,t}$ converges to $E[\mathbf{n}_{\mathcal{C}}]$ as K and t go to infinity. The plots show relative error $||\bar{\mathbf{n}}_{\mathcal{C}}^t - M\boldsymbol{\mu}_{\mathcal{C}}||/||M\boldsymbol{\mu}_{\mathcal{C}}||$ as a function of t for $K = 30$.

References

[1] V. Chvátal. *Linear Programming*. W.H. Freeman, New York, NY, 1983.