Journal of Machine Learning Research 7 (2006) 1743-1788 Submitted 11/05; Revised 4/06; Published 8/06

Learning Factor Graphs in Polynomial Time and Sample Complexity

Pieter Abbeel PABBEEL@CS.STANFORD.EDU
Daphne Koller KOLLER@CS.STANFORD.EDU
Andrew Y. Ng ANG@CS.STANFORD.EDU
Computer Science Department
Stanford University
Stanford, CA 94305, USA

Editor: Sanjoy Dasgupta

Abstract

We study the computational and sample complexity of parameter and structure learning in graphical models. Our main result shows that the class of factor graphs with bounded degree can be learned in polynomial time and from a polynomial number of training examples, assuming that the data is generated by a network in this class. This result covers both parameter estimation for a known network structure and structure learning. It implies as a corollary that we can learn factor graphs for both Bayesian networks and Markov networks of bounded degree, in polynomial time and sample complexity. Importantly, unlike standard maximum likelihood estimation algorithms, our method does not require inference in the underlying network, and so applies to networks where inference is intractable. We also show that the error of our learned model degrades gracefully when the generating distribution is not a member of the target class of networks. In addition to our main result, we show that the sample complexity of parameter learning in graphical models has an $O(1)$ dependence on the number of variables in the model when using the KL-divergence normalized by the number of variables as the performance criterion.[1]

Keywords: probabilistic graphical models, parameter and structure learning, factor graphs, Markov networks, Bayesian networks

1. Introduction

Graphical models are widely used to compactly represent structured probability distributions over (large) sets of random variables. Learning a graphical model from data is important for many applications. This learning problem can vary along several axes, including whether the data is fully or partially observed, and whether the structure of the network is given or needs to be learned from data.

In this paper, we focus on the problem of learning both network structure and parameters from fully observable data, restricting attention to discrete probability distributions over finite sets. We focus on the problem of learning a factor graph representation (Kschischang et al., 2001) of the distribution. Factor graphs subsume both Bayesian networks and Markov networks, in that every Bayesian network or Markov network can be written as a factor graph of (essentially) the same size.[2]

1. A preliminary version of some of this work was reported in Abbeel et al. (2005).
2. The factor graph corresponding to either a Bayesian network or a Markov network can be constructed in linear time (as a function of the size of the original network). See, for example, Kschischang et al. (2001), and Yedidia et al.

We provide a new parameterization of factor graph distributions, which forms the basis for our results. In this new parameterization, every factor is written as a product of probabilities over the variables in the factor and its neighbors. We will refer to such subsets of variables as "local subsets of variables." These local subsets of variables are of size at most d^2 for factor graphs of bounded degree d. Thus, for factor graphs of bounded degree d, the probabilities appearing in our new parameterization are over at most d^2 variables and can be estimated efficiently from training examples.[3] Hence this new parameterization naturally leads to an algorithm that solves the *parameter learning* problem *in closed-form* by estimating the probabilities over these local subsets of variables from training examples. We show that our closed-form estimation procedure results in a good estimate of the true distribution. More specifically, for factor graphs of bounded degree, if the generating distribution falls into the target class, we show that our estimation procedure returns an accurate solution—one of low KL-divergence from the true distribution—given a *polynomial number of training examples*.

In contrast to our new parameterization, the factors in a factor graph (or a Markov network) are typically considered to have no probabilistic interpretation at all. One exception is the canonical parameterization used in the Hammersley-Clifford theorem for Markov networks (Hammersley and Clifford, 1971; Besag, 1974b). The Hammersley-Clifford canonical parameterization expresses the distribution as a product of probabilities over *all* variables. However, the number of different instantiations is exponential in the number of variables. Therefore such probabilities over all variables cannot be estimated accurately from a small number of training examples. As a consequence the Hammersley-Clifford canonical parameterization is not suited for parameter learning.

Our closed-form parameter learning algorithm is the first polynomial-time and polynomial sample-complexity parameter learning algorithm for factor graphs of bounded degree, and thereby for Markov networks of bounded degree. In contrast, we do not know how to do maximum likelihood (ML) estimation in Markov networks or factor graphs without evaluating the likelihood. Evaluating the likelihood is equivalent to evaluating the partition function. Evaluating the partition function is known to be NP-hard, both exactly and approximately (Jerrum and Sinclair, 1993; Barahona, 1982). Indeed, all known exact algorithms grow exponentially in the tree-width of the graph, making the computation of the partition function intractable for many, even moderately sized, factor graphs. (See, for example, Cowell et al., 1999, for more details on such exact algorithms.) For example, n by n grids over binary variables (which have degree bounded by 4, independently of n) have tree-width n and the computational complexity of known algorithms for computing the partition function (and thus of known ML algorithms) is $O(2^n)$.

We analyze the sample complexity of parameter learning as a function of the number of variables in the network. We show that (under some mild assumptions) the sample complexity of parameter learning in graphical models has on $O(1)$ dependence on the number of variables in the graphical model when using KL-divergence normalized by the number of variables as the performance criterion. This result is important since it gives theoretical support for the common practice of learning large graphical models from a relatively small number of training examples. More specifically, the number of training examples can be much smaller than the number of parameters when learning large graphical models.

(2001), for more details on the equivalence and conversion between factor graphs, Bayesian networks and Markov networks.

3. For a pairwise Markov network with degree of the undirected graph bounded by d, the local subsets are of size at most $2d$.

Building on our closed-form parameter learning algorithm, we provide an algorithm for learning not only the parameters, but also the *structure*. In our new parameterization, factors that are not present in the distribution can be computed in the same way from local probabilities as factors that are present in the distribution. As will become clear later, a key property of our new parameterization is that the factors not present in the distribution have all entries equal to one. This gives a very simple test to decide whether or not a factor is present in the distribution. Thus no iterative search procedure—as is common for most structure learning algorithms—is needed. However, to compute all the factors from local probabilities, we need to know which variables are its neighbors. So to complete the structure learning algorithm, we need to show how to find each factor's neighbors. We show that local independence tests can be used to find the neighbors of each factor. Since local independence tests use statistics over a small number of variables only, the neighbors can be found efficiently from a small number of training examples.

Our structure learning algorithm provides the first polynomial-time and polynomial sample-complexity structure learning algorithm for factor graphs, and thereby for Markov networks. Note that our algorithm applies to any factor graph of bounded degree, including those (such as grids) where inference is intractable.

We also show that our algorithms degrade gracefully, in that they return reasonable answers even when the underlying distribution does not come exactly from the target class of networks.

We note that the proposed algorithms are unlikely to be useful in practice in their current form. The structure learning algorithm does an exhaustive enumeration over the possible neighbor sets of factors in the factor graph, a process which is—although polynomial—generally infeasible even in moderately sized networks. Both the parameter and the structure learning algorithm do not make good use of all the available data. Nevertheless, the techniques used in our analysis open new avenues towards efficient parameter and structure learning in undirected, intractable models.

The remainder of this paper is organized as follows. Section 2 provides necessary background about Gibbs distributions, the factor graph associated with a Gibbs distribution, Markov blankets and the Hammersley-Clifford canonical parameterization. In its original form, the Hammersley-Clifford theorem applies to Markov networks only. We provide an extension that applies to factor graphs. In Section 3, building on the canonical parameterization for factor graphs, we derive our novel parameterization, which forms the basis of our parameter estimation algorithm. We present our algorithm and provide formal running time and sample complexity guarantees. We conclude the section with an in-depth analysis of the relationship between the sample complexity and the number of random variables. In Section 4, we present our structure learning algorithm, and its formal guarantees. Section 5 discusses related work. For clarity of exposition, we provide the complete proofs of all theorems and propositions in the appendix.

Table 1 gives an overview of the notation we use throughout this paper.

2. Preliminaries

In this section we first introduce Gibbs distributions, the factor graph associated with a Gibbs distribution, Markov blankets and the canonical parameterization. Then we present an extension of the Hammersley-Clifford theorem—which in its original form only applies to Markov networks—to factor graphs. Throughout the paper we restrict attention to discrete probability distributions over finite sets.

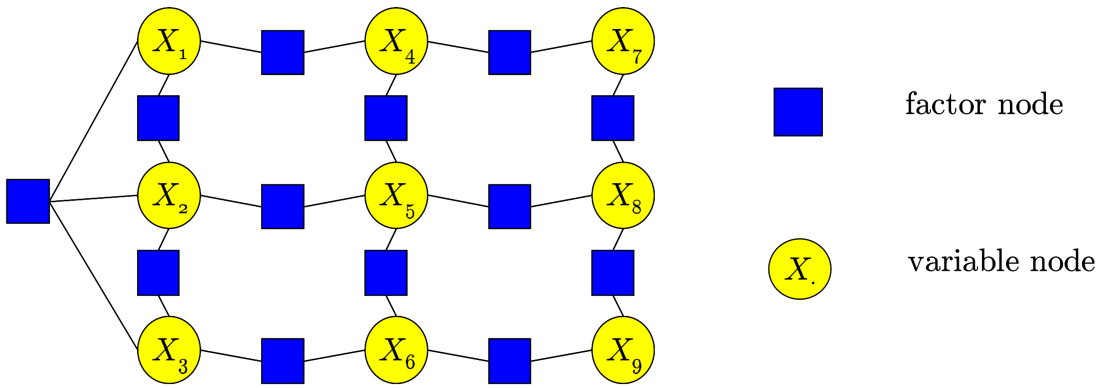

Figure 1: Example factor graph.

2.1 Gibbs Distributions

The probability distributions we consider are referred to as Gibbs distributions.

Definition 1 (Gibbs distribution) *A* factor f *with* scope[4] $\mathbf{D}$ *is a mapping from* $\mathrm{val}(\mathbf{D})$ *to* $\mathbb{R}^+$. *A* Gibbs *distribution* P *over a set of random variables* $\mathcal{X} = \{X_1, \ldots, X_n\}$ *is associated with a set of factors* $\{f_j\}_{j=1}^J$ *with scopes* $\{\mathbf{C}_j\}_{j=1}^J$*, such that*

$$P(X_1 = x_1, \ldots, X_n = x_n) = \frac{1}{Z} \prod_{j=1}^{J} f_j(\mathbf{C}_j[x_1, \ldots, x_n]).$$

The normalizing constant Z *is the* partition function.

The *factor graph* associated with a Gibbs distribution is a bipartite graph whose nodes correspond to variables and factors, with an edge between a variable X and a factor f_j if the scope of f_j contains X. There is one-to-one correspondence between factor graphs and the sets of scopes. Figure 1 gives an example of a factor graph. Here the Gibbs distribution is over the variables $X_1, \cdots, X_9$, which are represented by circles in the factor graph. The factors are represented by squares and have the following respective scopes: $\{X_1,X_2,X_3\}$, $\{X_1,X_2\}$, $\{X_2,X_3\}$, $\{X_1,X_4\}$, $\{X_2,X_5\}$, $\{X_3,X_6\}$, $\{X_4,X_5\}$, $\{X_5,X_6\}$, $\{X_4,X_7\}$, $\{X_5,X_8\}$, $\{X_7,X_9\}$, $\{X_7,X_8\}$, $\{X_8,X_9\}$. The corresponding Gibbs distribution is given by

$$P(X_1 = x_1, \cdots, X_9 = x_9) = \frac{1}{Z} f_{\{X_1,X_2,X_3\}}(x_1,x_2,x_3) f_{\{X_1,X_2\}}(x_1,x_2) \cdots f_{\{X_8,X_9\}}(x_8,x_9).$$

A Gibbs distribution also induces a Markov network—an undirected graph whose nodes correspond to the random variables $\mathcal{X}$ and where there is an edge between two variables if there is a factor in which they both participate. The set of scopes uniquely determines the structure of the Markov network, but several different sets of scopes can result in the same Markov network. For example, a fully connected Markov network can correspond both to a Gibbs distribution with $\binom{n}{2}$ factors over pairs of variables, and to a distribution with a factor which is a joint distribution over $\mathcal{X}$. We will

4. A function has *scope* $\mathbf{X}$ if its domain is $\mathrm{val}(\mathbf{X})$, the set of possible instantiations of the set of random variables $\mathbf{X}$.

use the more precise factor graph representation in this paper. Our results are easily translated into results for Markov networks.

Definition 2 (Markov blanket) *Let a set of scopes $\mathcal{C} = \{\mathbf{C}_j\}_{j=1}^J$ be given. The* Markov blanket *of a set of random variables $\mathbf{D} \subseteq \mathcal{X}$ is defined as*

$$\mathrm{MB}(\mathbf{D}) = \cup\{\mathbf{C}_j : \mathbf{C}_j \in \mathcal{C}\ ,\ \mathbf{C}_j \cap \mathbf{D} \neq \emptyset\} - \mathbf{D}.$$

Thus, the Markov blanket of a set of variables $\mathbf{D}$ is the minimal set of variables that separates $\mathbf{D}$ from the other variables in the factor graph. For the factor graph distribution of Figure 1 we have, for example, $\mathrm{MB}(\{X_1\}) = \{X_2, X_3, X_4\}$, $\mathrm{MB}(\{X_1, X_2\}) = \{X_3, X_4, X_5\}$, and $\mathrm{MB}(\{X_5\}) = \{X_2, X_4, X_6, X_8\}$.

For any Gibbs distribution, we have, for any subset of random variables $\mathbf{D}$, that

$$\mathbf{D} \perp \mathcal{X} - \mathbf{D} - \mathrm{MB}(\mathbf{D}) \ \mid\ \mathrm{MB}(\mathbf{D}), \tag{1}$$

or in words: given its Markov blanket $\mathrm{MB}(\mathbf{D})$, the set of variables $\mathbf{D}$ is independent of all other variables $\mathcal{X} - \mathbf{D} - \mathrm{MB}(\mathbf{D})$.[5]

A standard assumption for a Gibbs distribution, which is critical for identifying its structure (see Lauritzen, 1996, Ch. 3), is that the distribution be *positive*—all of its entries be non-zero. Our results use a quantitative measure for how positive P is. Let $\gamma = \min_{\mathbf{x},i} P(X_i = x_i | \mathcal{X}_{-i} = \mathbf{x}_{-i})$, where the $-i$ subscript denotes all entries but entry i. Note that, if we have a fixed bound on the number of factors in which a variable can participate, a fixed bound on the domain size for each variable, and a fixed bound on how skewed each factor is (more specifically a bound on the ratio of its lowest and highest entries), we are guaranteed a bound on γ that is independent of the number n of variables in the network. Thus, under these assumptions, our sample complexity results, which are expressed as a function of γ, have *no* hidden dependence on the number of variables n. In contrast, $\tilde{\gamma} = \min_{\mathbf{x}} P(\mathbf{X} = \mathbf{x})$ generally has an exponential dependence on n. For example, if we have n independent and identically distributed (i.i.d.) Bernoulli($\frac{1}{2}$) random variables, then $\gamma = \frac{1}{2}$ (independent of n) but $\tilde{\gamma} = \frac{1}{2^n}$.

2.2 The Canonical Parameterization

A Gibbs distribution is generally over-parameterized relative to the structure of the underlying factor graph, in that a continuum of possible parameterizations over the graph can all encode the same distribution. The *canonical parameterization* (Hammersley and Clifford, 1971; Besag, 1974b) provides one specific choice of parameterization for a Gibbs distribution, with some nice properties (see below). The canonical parameterization forms the basis for the Hammersley-Clifford theorem, which asserts that any distribution that satisfies the independence assumptions encoded by a Markov network can be represented as a Gibbs distribution with factors corresponding to each of the cliques in the Markov network. In its original formulation, the canonical distribution is defined for Gibbs distributions over Markov networks. We use a more refined parameterization, defined at the factor level; results at the clique level (or, equivalently, results for Markov networks) are trivial corollaries.

The canonical parameterization is defined relative to an arbitrary (but fixed) set of "default" assignments $\bar{\mathbf{x}} = (\bar{x}_1, \ldots, \bar{x}_n)$. Let any subset of variables $\mathbf{D} = \langle X_{i_1}, \ldots, X_{i_{|\mathbf{D}|}} \rangle$, and any assignment

5. By $\mathbf{X} \perp \mathbf{Y}$ we denote that $\mathbf{X}$ is independent of $\mathbf{Y}$. By $\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z}$ we denote that $\mathbf{X}$ is conditionally independent of $\mathbf{Y}$ given $\mathbf{Z}$.

$\mathbf{d} = \langle x_{i_1}, \ldots, x_{i_{|\mathbf{D}|}} \rangle$ be given. Let any $\mathbf{U} \subseteq \mathbf{D}$ be given. We define $\sigma_{\cdot}[\cdot]$ such that for all $i \in \{1, \ldots, n\}$:

$$(\sigma_{\mathbf{U}}[\mathbf{d}])_i = \begin{cases} x_i & \text{if } X_i \in \mathbf{U}, \\ \bar{x}_i & \text{if } X_i \notin \mathbf{U}. \end{cases}$$

In words, $\sigma_{\mathbf{U}}[\mathbf{d}]$ keeps the assignments to the variables in $\mathbf{U}$ as specified in $\mathbf{d}$, and augments it to form a full assignment using the default values in $\bar{\mathbf{x}}$. Note that the assignments to variables outside $\mathbf{U}$ are always ignored, and replaced with their default values. Thus, the scope of $\sigma_{\mathbf{U}}[\cdot]$ is always $\mathbf{U}$.

Let P be a positive Gibbs distribution.The *canonical factor* for $\mathbf{D} \subseteq \mathcal{X}$ is defined as follows:

$$f^*_{\mathbf{D}}(\mathbf{d}) = \exp\left(\textstyle\sum_{\mathbf{U} \subseteq \mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \log P(\sigma_{\mathbf{U}}[\mathbf{d}])\right). \tag{2}$$

The sum is over all subsets of $\mathbf{D}$, including $\mathbf{D}$ itself and the empty set $\emptyset$.

The following theorem extends the Hammersley-Clifford theorem (which applies to Markov networks) to factor graphs.

Theorem 3 *Let P be a positive Gibbs distribution with factor scopes $\{\mathbf{C}_j\}_{j=1}^J$. Let $\{\mathbf{C}^*_j\}_{j=1}^{J^*} = \cup_{j=1}^J 2^{\mathbf{C}_j} - \emptyset$ (where $2^{\mathbf{X}}$ is the power set of $\mathbf{X}$—the set of all of its subsets). Then*

$$P(\mathbf{x}) = P(\bar{\mathbf{x}}) \textstyle\prod_{j=1}^{J^*} f^*_{\mathbf{C}^*_j}(\mathbf{c}^*_j),$$

where $\mathbf{c}^*_j$ is the instantiation of $\mathbf{C}^*_j$ consistent with $\mathbf{x}$.

The proof is in the appendix.

The parameterization of P using the canonical factors $\{f^*_{\mathbf{C}^*_j}\}_{j=1}^{J^*}$ is called the *canonical parameterization* of P. Although typically $J^* > J$, the additional factors are all subfactors of the original factors. Note that first transforming a factor graph into a Markov network and then applying the Hammersley-Clifford theorem to the Markov network generally results in a significantly less sparse canonical parameterization than the canonical parameterization from Theorem 3.

We now give an example to clarify the definition of canonical factors and canonical parameterization.

Example 1 *Consider again the factor graph of Figure 1. Assume we take the fixed assignment to be all zeros, namely we have $\bar{x}_1 = 0, \bar{x}_2 = 0, \cdots, \bar{x}_9 = 0$. Then the canonical factor $f^*_{\{X_1,X_2\}}$ over the variables X_1, X_2 instantiated to x_1, x_2 is given by*

$$\begin{aligned} \log f^*_{\{X_1,X_2\}}(x_1,x_2) \quad = \quad & \log P(X_1 = x_1, X_2 = x_2, X_3 = 0, X_4 = 0, \cdots, X_9 = 0) \\ & - \log P(X_1 = 0, X_2 = x_2, X_3 = 0, X_4 = 0, \cdots, X_9 = 0) \\ & - \log P(X_1 = x_1, X_2 = 0, X_3 = 0, X_4 = 0, \cdots, X_9 = 0) \\ & + \log P(X_1 = 0, X_2 = 0, X_3 = 0, X_4 = 0, \cdots, X_9 = 0). \end{aligned} \tag{3}$$

So to compute the canonical factor, we start with the joint instantiation of the factor variables $\{X_1, X_2\}$ with all other variables $\{X_3, \cdots, X_9\}$ set to their default instantiations. Then we subtract out the instantiations for which one of the factor variables is changed to its default instantiation. Crudely speaking, we subtract out the interactions that are already captured by a canonical factor over a smaller set of variables. Then we adjust for double counting by adding back in the instantiation where both factor variables have been set to their default instantiation.

Similarly, the canonical factor $f^*_{\{X_1,X_2,X_3\}}$ over the variables X_1,X_2,X_3 instantiated to x_1,x_2,x_3 is given by

$$\begin{aligned}\log f^*_{\{X_1,X_2,X_3\}}(x_1,x_2,x_3) \;=\;& \log P(X_1=x_1,X_2=x_2,X_3=x_3,X_4=0,\cdots,X_9=0)\\ &-\log P(X_1=0,X_2=x_2,X_3=x_3,X_4=0,\cdots,X_9=0)\\ &-\log P(X_1=x_1,X_2=0,X_3=x_3,X_4=0,\cdots,X_9=0)\\ &-\log P(X_1=x_1,X_2=x_2,X_3=0,X_4=0,\cdots,X_9=0)\\ &+\log P(X_1=0,X_2=0,X_3=x_3,X_4=0,\cdots,X_9=0)\\ &+\log P(X_1=0,X_2=x_2,X_3=0,X_4=0,\cdots,X_9=0)\\ &+\log P(X_1=x_1,X_2=0,X_3=0,X_4=0,\cdots,X_9=0)\\ &-\log P(X_1=0,X_2=0,X_3=0,X_4=0,\cdots,X_9=0).\end{aligned}$$

The canonical factor over just the variable X_1 instantiated to x_1 is given by

$$\begin{aligned}\log f^*_{\{X_1\}}(x_1) \;=\;& \log P(X_1=x_1,X_2=0,X_3=0,X_4=0,\cdots,X_9=0)\\ &-\log P(X_1=0,X_2=0,X_3=0,X_4=0,\cdots,X_9=0).\end{aligned}$$

Theorem 3 applied to our example gives the following expression for the probability distribution:

$$\begin{aligned}P(X_1=x_1,\cdots,X_9=x_9) \;=\;& P(X_1=0,\cdots,X_9=0)\\ &\times f^*_{\{X_1,X_2,X_3\}}(x_1,x_2,x_3)\\ &\times f^*_{\{X_1,X_2\}}(x_1,x_2)f^*_{\{X_2,X_3\}}(x_2,x_3)\cdots f^*_{\{X_8,X_9\}}(x_8,x_9)\\ &\times f^*_{\{X_1\}}(x_1)f^*_{\{X_2\}}(x_2)\cdots f^*_{\{X_9\}}(x_9)\\ =\;& \frac{1}{Z}\\ &\times f^*_{\{X_1,X_2,X_3\}}(x_1,x_2,x_3)\\ &\times f^*_{\{X_1,X_2\}}(x_1,x_2)f^*_{\{X_2,X_3\}}(x_2,x_3)\cdots f^*_{\{X_8,X_9\}}(x_8,x_9)\\ &\times f^*_{\{X_1\}}(x_1)f^*_{\{X_2\}}(x_2)\cdots f^*_{\{X_9\}}(x_9).\end{aligned} \tag{4}$$

3. Parameter Estimation

In this section we first introduce the parameter estimation ideas informally by expanding on Example 1. Then we formally introduce the key idea of Markov blanket canonical factors, which give a parameterization of a factor graph distribution only in terms of local probabilities. This new parameterization directly results in the proposed parameter estimation algorithm. We analyze the algorithm's computational and sample complexity. In addition, we show an $O(1)$ dependence on the number of variables in the network for the sample complexity when using the KL-divergence normalized by the number of variables in the network as performance criterion.

3.1 Parameter Estimation by Example

Consider the problem of estimating the parameters of the distribution in Figure 1 from training examples. From Eqn. (4) we have that it is sufficient to estimate all the canonical factors. Each canonical factor is expressed in terms of probabilities. So one could estimate the canonical factors (and thus the distribution) in closed-form by estimating these probabilities from data. Unfortunately the probabilities appearing in the canonical factors are over *full joint instantiations of all variables*. As a consequence, these probabilities can not be estimated accurately from a small amount of data.

However, we will now consider the factor $f^*_{\{X_1,X_2\}}$ more carefully and show it can be estimated from probabilities over small subsets of the variables only. The factor $f^*_{\{X_1,X_2\}}$ contains an equal number of terms with positive and negative sign. For the sum of two such terms, we now derive a novel expression which contains local probabilities only (instead of probabilities of full joint instantiations of all variables).

$$\begin{aligned}
& \log P(X_1=x_1,X_2=x_2,X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
& -\log P(X_1=x_1,X_2=0,X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
= \;& \log P(X_1=x_1,X_2=x_2|X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
& +\log P(X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
& -\log P(X_1=x_1,X_2=0|X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
& -\log P(X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
= \;& \log P(X_1=x_1,X_2=x_2|X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
& -\log P(X_1=x_1,X_2=0|X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0) \\
= \;& \log P(X_1=x_1,X_2=x_2|\mathrm{MB}(\{X_1,X_2\})=\vec{0}) \\
& -\log P(X_1=x_1,X_2=0|\mathrm{MB}(\{X_1,X_2\})=\vec{0}) \\
= \;& \log P(X_1=x_1,X_2=x_2|X_3=0,X_4=0,X_5=0) \\
& -\log P(X_1=x_1,X_2=0|X_3=0,X_4=0,X_5=0).
\end{aligned} \tag{5}$$

Here we used in order: the definition of conditional probability; same terms with opposite sign cancel; conditioning on the Markov blanket is equivalent to conditioning on all other variables; $\mathrm{MB}(\{X_1,X_2\})=\{X_3,X_4,X_5\}$ in our example.

The last expression in Eqn. (5) contains local probabilities only, which can be estimated accurately from a small number of training examples. Using a similar reasoning as above for the other two terms of the factor $f^*_{\{X_1,X_2\}}$, we get the following expression for $f^*_{\{X_1,X_2\}}$, which contains local probabilities only:

$$\begin{aligned}
\log f^*_{\{X_1,X_2\}}(x_1,x_2) \;=\; & \log P(X_1=x_1,X_2=x_2|X_3=0,X_4=0,X_5=0) \\
& -\log P(X_1=x_1,X_2=0|X_3=0,X_4=0,X_5=0) \\
& -\log P(X_1=0,X_2=x_2|X_3=0,X_4=0,X_5=0) \\
& +\log P(X_1=0,X_2=0|X_3=0,X_4=0,X_5=0) \\
=\; & \log f^*_{\{X_1,X_2\}|\{X_3,X_4,X_5\}}(x_1,x_2).
\end{aligned} \tag{6}$$

The last line defines $f^*_{\{X_1,X_2\}|\{X_3,X_4,X_5\}}(x_1,x_2)$ (which we refer to as the Markov blanket canonical factor for $\{X_1,X_2\}$). Although $f^*_{\{X_1,X_2\}}(x_1,x_2)=f^*_{\{X_1,X_2\}|\{X_3,X_4,X_5\}}(x_1,x_2)$ when exact probabilities are used, we use different notation to explicitly distinguish how they are computed from probabilities. The Markov blanket canonical factor $f^*_{\{X_1,X_2\}|\{X_3,X_4,X_5\}}(x_1,x_2)$ is computed from local probabilities as given in Eqn. (6). The (original) canonical factor $f^*_{\{X_1,X_2\}}(x_1,x_2)$ is computed from probabilities over full joint instantiations as given in Eqn. (3).

Similarly, the other canonical factors have equivalent Markov blanket canonical factors which involve local probabilities only. This gives us an efficient closed-form parameter estimation algorithm for our example. In the next few sections we formalize this idea for general factor graphs and analyze the computational and sample complexity.

3.2 Markov Blanket Canonical Factors

Considering the definition of the canonical parameters, we note that all of the terms in Eqn. (2) can be estimated from empirical data using simple counts, without requiring inference over the network. Thus, it appears that we can use the canonical parameterization as the basis for our parameter estimation algorithm. However, as written, this estimation process is statistically infeasible, as the terms in Eqn. (2) are probabilities over full instantiations of all variables, which can never be estimated from a reasonable number of training examples.

We now generalize our observation from the example in the previous section: namely, that we can express the canonical factors using only probabilities over much smaller instantiations—those corresponding to a factor and its Markov blanket. Let $\mathbf{D} = \langle X_{i_1}, \ldots, X_{i_{|\mathbf{D}|}} \rangle$ be any subset of variables, and $\mathbf{d} = \langle x_{i_1}, \ldots, x_{i_{|\mathbf{D}|}} \rangle$ be any assignment to $\mathbf{D}$. For any $\mathbf{U} \subseteq \mathbf{D}$, we define $\sigma_{\mathbf{U}:\mathbf{D}}[\mathbf{d}]$ to be the restriction of the full instantiation $\sigma_{\mathbf{U}}[\mathbf{d}]$ of all variables in $\mathcal{X}$ to the corresponding instantiation of the subset $\mathbf{D}$. In other words, $\sigma_{\mathbf{U}:\mathbf{D}}[\mathbf{d}]$ keeps the assignments to the variables in $\mathbf{U}$ as specified in $\mathbf{d}$, and changes the assignment to the variables in $\mathbf{D}-\mathbf{U}$ to the default values in $\bar{\mathbf{x}}$. Let $\mathbf{D} \subseteq \mathcal{X}$ and $\mathbf{Y} \subseteq \mathcal{X} - \mathbf{D}$. Then the factor $f^*_{\mathbf{D}|\mathbf{Y}}$ over the variables in $\mathbf{D}$ is defined as follows:

$$f^*_{\mathbf{D}|\mathbf{Y}}(\mathbf{d}) = \exp\left(\textstyle\sum_{\mathbf{U}\subseteq\mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \log P(\sigma_{\mathbf{U}:\mathbf{D}}[\mathbf{d}] \,|\, \mathbf{Y} = \bar{\mathbf{y}})\right), \tag{7}$$

where the sum is over all subsets of $\mathbf{D}$, including $\mathbf{D}$ itself and the empty set $\emptyset$.

For example, we have that $f^*_{\{X_1,X_2\}|\{X_3,X_4,X_5\}}$ of the factor graph in Figure 1 is given by Eqn. (6) in the previous section.

The following proposition shows an equivalence between the factors computed using Eqn. (2) and Eqn. (7).

Proposition 4 *Let P be a positive Gibbs distribution with factor scopes $\{\mathbf{C}_j\}_{j=1}^J$, and $\{\mathbf{C}^*_j\}_{j=1}^{J^*}$ as above (i.e., $\{\mathbf{C}^*_j\}_{j=1}^{J^*} = \cup_{j=1}^J 2^{\mathbf{C}_j} - \emptyset$). Then for any $\mathbf{D} \subseteq \mathcal{X}$, we have:*

$$f^*_{\mathbf{D}} = f^*_{\mathbf{D}|\mathcal{X}-\mathbf{D}} = f^*_{\mathbf{D}|\mathrm{MB}(\mathbf{D})}, \tag{8}$$

and (as a direct consequence)

$$\begin{aligned} P(\mathbf{x}) &= P(\bar{\mathbf{x}}) \textstyle\prod_{j=1}^{J^*} f^*_{\mathbf{C}^*_j|\mathcal{X}-\mathbf{C}^*_j}(\mathbf{c}^*_j) && (9)\\ &= P(\bar{\mathbf{x}}) \textstyle\prod_{j=1}^{J^*} f^*_{\mathbf{C}^*_j|\mathrm{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j), && (10)\end{aligned}$$

where $\mathbf{c}^*_j$ is the instantiation of $\mathbf{C}^*_j$ consistent with $\mathbf{x}$.

Proposition 4 shows that we can compute the canonical parameterization factors using probabilities over factor scopes and their Markov blankets only. From a sample complexity point of view, this is a significant improvement over the standard definition which uses joint instantiations over all variables. Using Eqn. (7) we can expand the Markov blanket canonical factors in Proposition 4 and we see that *any factor graph distribution can be parameterized as a product of local probabilities only*.

$X,Y,\ldots$	random variables
$x,y,\ldots$	instantiations of the random variables
$\mathbf{X},\mathbf{Y},\ldots$	sets of random variables
$\mathbf{x},\mathbf{y},\ldots$	instantiations of sets of random variables
$\mathrm{val}(X)$	set of values the variable X can take
$\mathbf{D}[\mathbf{x}]$	instantiation of $\mathbf{D}$ consistent with $\mathbf{x}$ (abbreviated as $\mathbf{d}$ when no ambiguity is possible)
$\mathbf{X} \perp \mathbf{Y}$	$\mathbf{X}$ is independent of $\mathbf{Y}$
$\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z}$	$\mathbf{X}$ is conditionally independent of $\mathbf{Y}$ given $\mathbf{Z}$
f	factor
P	positive Gibbs distribution over a set of random variables $\mathcal{X} = \langle X_1,\ldots,X_n\rangle$
$\{f_j\}_{j=1}^J$	factors of P
$\{\mathbf{C}_j\}_{j=1}^J$	scopes of factors of P
$\hat{P}$	empirical (sample) distribution
$\tilde{P}$	distribution returned by learning algorithm
$f^*_{\cdot}$	canonical factor as defined in Eqn. (2)
$f^*_{\cdot\mid\cdot}$	canonical factor as defined in Eqn. (7)
$\hat{f}^*_{\cdot\mid\cdot}$	canonical factor as defined in Eqn. (7), but using the empirical distribution $\hat{P}$
$\mathrm{MB}(\mathbf{D})$	Markov blanket of $\mathbf{D}$
k	$\max_j\lvert\mathbf{C}_j\rvert$
γ	$\min_{\mathbf{x},i} P(X_i = x_i\lvert\mathcal{X}_{-i} = \mathbf{x}_{-i})$
v	$\max_i\lvert\mathrm{val}(\mathrm{X_i})\rvert$
b	$\max_j\lvert\mathrm{MB}(\mathbf{C}_j)\rvert$
m	number of training examples
$D(\cdot\Vert\cdot)$	KL-divergence, $D(P\Vert Q) = \sum_{\mathbf{x}\in\mathrm{val}\mathcal{X}} P(\mathbf{x})\log\frac{P(\mathbf{x})}{Q(\mathbf{x})}$
$\mathcal{C}$	the set of candidate factor scopes for the structure learning algorithm, Factor-Graph-Structure-Learn ($\mathcal{C} = \{\mathbf{C}^*_j : \mathbf{C}^*_j \subseteq \mathcal{X}, \mathbf{C}^*_j \neq \emptyset, \lvert\mathbf{C}^*_j\rvert \leq k\}$)

Table 1: Notational conventions.

3.3 Parameter Estimation Algorithm

Based on the parameterization above, we propose the following Factor-Graph-Parameter-Learn algorithm. The algorithm takes as inputs: the scopes of the factors $\{\mathbf{C}_j\}_{j=1}^J$, training examples $\{\mathbf{x}^{(i)}\}_{i=1}^m$, a baseline instantiation $\bar{\mathbf{x}}$. Then for $\{\mathbf{C}^*_j\}_{j=1}^{J^*}$ as above (i.e., $\{\mathbf{C}^*_j\}_{j=1}^{J^*} = \cup_{j=1}^J 2^{\mathbf{C}_j} - \emptyset$), Factor-Graph-Parameter-Learn does the following:

- Compute the estimates of the canonical factors $\{\hat{f}^*_{\mathbf{C}^*_j\mid\mathrm{MB}(\mathbf{C}^*_j)}\}_{j=1}^{J^*}$ as in Eqn. (7), but using the empirical estimates based on the training examples.
- Return the probability distribution $\tilde{P}(\mathbf{x}) \propto \prod_{j=1}^{J^*}\hat{f}^*_{\mathbf{C}^*_j\mid\mathrm{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)$.

Theorem 5 (Parameter learning: computational complexity) *The running time of the* Factor-Graph-Parameter-Learn *algorithm is in* $O(m2^k J(k+b) + 2^{2k} J v^k)$.[6]

The proof is given in the appendix.

Note the representation of the factor graph distribution is $\Omega(Jv^k)$, thus exponential dependence on k is unavoidable for any algorithm. More importantly, there is no dependence on the running time of evaluating the partition function. On the other hand, all currently known maximum likelihood estimation algorithms require evaluating the partition function, which is known to be NP-hard, both exactly and approximately (Jerrum and Sinclair, 1993; Barahona, 1982).

3.4 Sample Complexity

We now analyze the sample complexity of the Factor-Graph-Parameter-Learn algorithm, showing that it returns a distribution that is a good approximation of the true distribution when given only a "small" number of training examples. We will use the sum of KL-divergences $D(P\|\tilde{P}) + D(\tilde{P}\|P)$ to measure how well the distribution $\tilde{P}$ approximates the distribution P.[7]

Theorem 6 (Parameter learning: sample complexity) *Let any* $\varepsilon, \delta > 0$ *be given. Let* Factor-Graph-Parameter-Learn *be given (a) m training examples* $\{\mathbf{x}^{(i)}\}_{i=1}^m$ *drawn i.i.d. from a distribution P and (b) the factor graph structure according to which the distribution P factors. Let* $\tilde{P}$ *be the probability distribution returned by* Factor-Graph-Parameter-Learn. *Then, we have that, for*

$$D(P\|\tilde{P}) + D(\tilde{P}\|P) \leq J\varepsilon$$

to hold with probability at least $1-\delta$*, it suffices that the number of training examples m satisfies:*

$$m \geq \left(1 + \frac{\varepsilon}{2^{2k+2}}\right)^2 \frac{2^{4k+3}}{\gamma^{2k+2b}\varepsilon^2} \log \frac{2^{k+2} J v^{k+b}}{\delta}. \tag{11}$$

A complete proof is given in the appendix.

Theorem 6 shows that—assuming the true distribution P factors according to the given structure—Factor-Graph-Parameter-Learn returns a distribution that is $J\varepsilon$-close in KL-divergence. The sample complexity scales exponentially in the maximum number of variables per factor k, and polynomially in $\frac{1}{\varepsilon}, \frac{1}{\gamma}$.

The error in the KL-divergence grows linearly with the number of factors J. This is a consequence of the fact that the number of terms in the distributions is equal to the number of factors J, and each term can accrue an error. We can obtain a more refined analysis if we eliminate this dependence by considering the KL-divergence normalized by the number of variables, $D_n(P\|\tilde{P}) = \frac{1}{n} D(P\|\tilde{P})$. We return to this topic in Section 3.5.

We now sketch the proof idea. The Markov blanket canonical factors are a product of local conditional probabilities. These local conditional probabilities can be estimated accurately from a "small" number of training examples. Thus the Markov blanket canonical factors can be estimated accurately from a small number of training examples. Thus the factor graph distribution—which is just a product of the Markov canonical factors—can be estimated accurately from a small number of training examples.

6. The upper bound is based on a very naive implementation's running time. It assumes that operations on numbers (such as reading, writing, adding, etc.) take constant time.

7. $D(P\|Q) = \sum_{\mathbf{x} \in \mathrm{val}X} P(\mathbf{x}) \log \frac{P(\mathbf{x})}{Q(\mathbf{x})}$.

Theorem 6 considers the case when P factors according to the given structure. The following theorem shows that our error degrades gracefully even if the training examples are generated by a distribution Q that does not factor according to the given structure.

Theorem 7 (Parameter learning: graceful degradation) *Let any $\varepsilon, \delta > 0$ be given. Let $\{\mathbf{x}^{(i)}\}_{i=1}^m$ be i.i.d. samples from a distribution Q. Let MB and $\widehat{\mathrm{MB}}$ be the Markov blankets according to the distribution Q and the given structure respectively. Let $\{f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}\}_{j=1}^{\bar{J}}$ be the non-trivial Markov blanket canonical factors of Q (those factors with not all entries equal to one). Let $\{\mathbf{C}^*_j\}_{j=1}^{J^*}$ be the scopes of the canonical factors in the factor graph given to the algorithm. Let $\tilde{P}$ be the probability distribution returned by* Factor-Graph-Parameter-Learn. *Then we have that for*

$$D(Q\|\tilde{P}) + D(\tilde{P}\|Q) \leq J\varepsilon + 2\sum_{j:\mathbf{D}^*_j \notin \{\mathbf{C}^*_k\}_{k=1}^{J^*}} \max_{\mathbf{d}^*_j} \left|\log f^*_{\mathbf{D}^*_j}(\mathbf{d}^*_j)\right| + 2\sum_{j\,:\,\mathrm{MB}(\mathbf{C}^*_j)\neq\widehat{MB}(\mathbf{C}^*_j)} \max_{\mathbf{c}^*_j} \left|\log \frac{f^*_{\mathbf{C}^*_j|\mathrm{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)}{f^*_{\mathbf{C}^*_j|\widehat{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)}\right|$$

to hold with probability at least $1-\delta$, it suffices that the number of training examples m satisfies Eqn. (11) of Theorem 6.

Note the sample complexity depends on parameters $k = \max_j |\mathbf{C}^*_j|$ and $b = \max_j |\mathrm{MB}(\mathbf{C}^*_j)|$ of the given target structure (rather than the true structure). The graceful degradation result is important, as it shows that each canonical factor that is incorrectly captured by our target structure adds at most a constant (namely, $l2^{l+1}\log\frac{1}{\gamma}$ for an incorrectly captured factor over l variables) to our bound on the KL-divergence.[8] This constant can be large, so we discuss the actual error contribution in more detail. A canonical factor could be incorrectly captured when the corresponding factor scope is not included in the given structure. Canonical factors are designed so that a factor over a set of variables captures only the residual interactions between the variables in its scope, once all interactions between its subsets have been accounted for in other factors. Thus, canonical factors over large scopes are often close to the trivial all-ones factor in practice. Therefore, if our structure approximation is such that it only ignores some of the larger-scope factors, the error in the approximation may be quite limited. A canonical factor could also be incorrectly captured when the given structure does not have the correct Markov blanket for that factor. The resulting error depends on how good an approximation of the Markov blanket we do have. See Section 4 for more details on the error caused by incorrect Markov blankets.

3.5 Reducing the Dependence on Network Size

Our previous analysis showed a linear dependence of the sample complexity on the number of factors J in the network (for parameter learning). In a sense, this dependence is inevitable. To understand why, consider a distribution P defined by a set of n independent Bernoulli random variables $X_1,\ldots,X_n$, each with parameter 0.5. Assume that Q is an approximation to P, where the X_i are still independent, but have parameter 0.4999. Intuitively, a Bernoulli(0.4999) distribution is a very good

8. Each factor over l variables is a fraction of a product of 2^{l-1} conditional probabilities over another product of 2^{l-1} conditional probabilities. Recall that $\gamma = \min_{\mathbf{x},i} P(X_i = x_i|X_{-i} = \mathbf{x}_{-i}) > 0$, so we have that each conditional probability over l variables lies in the interval $[\gamma^l, 1]$. Thus we have for a factor over l variables that $\max_{\mathbf{d}^*_j}\left|\log f^*_{\mathbf{D}^*_j}(\mathbf{d}^*_j)\right| \leq \log\frac{1}{\gamma^{2^{l-1}}} = l2^{l-1}\log\frac{1}{\gamma}$. Similarly, we have that $\max_{\mathbf{c}^*_j}\left|\log\frac{f^*_{\mathbf{C}^*_j|\mathrm{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)}{f^*_{\mathbf{C}^*_j|\widehat{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)}\right| \leq l2^l\log\frac{1}{\gamma}$.

estimate of a Bernoulli(0.5); thus, for most applications, Q can safely be considered to be a very good estimate of P. However, the KL-divergence $D(P(X_{1:n})\|Q(X_{1:n})) = \sum_{i=1}^{n} D(P(X_i)\|Q(X_i)) = \Omega(n)$. Thus, if n is large, the KL divergence between P and Q would be large, even though Q is a good estimate for P. To remove such unintuitive scaling effects when studying the dependence on the number of variables, we can consider instead the normalized KL divergence criterion:

$$D_n(P(X_{1:n})\|Q(X_{1:n})) = \frac{1}{n}D(P(X_{1:n})\|Q(X_{1:n})).$$

As we now show, with a slight modification to the algorithm, we can achieve a bound of ε for our normalized KL-divergence while eliminating the logarithmic dependence on J in our sample complexity bound. Specifically, we can modify our algorithm so that it clips probability estimates $\in [0,\gamma^{k+b})$ to γ^{k+b}. The clipping procedure is motivated by the proof of Theorem 8 and effectively ensures that the KL-divergence is bounded.[9] Note that—since true probabilities which we are trying to estimate are never in the interval $[0,\gamma^{k+b})$—this change can only improve the estimates.[10]

For this slightly modified version of the algorithm, the following theorem shows the dependence on the size of the network is $O(1)$, which is tighter than the logarithmic dependence shown in Theorem 6.[11]

Theorem 8 (Parameter learning: size of the network) *Let any $\varepsilon,\delta > 0$ be given and fixed. Let $\{\mathbf{x}^{(i)}\}_{i=1}^{m}$ be i.i.d. samples from P. Let the domain size of each variable be fixed. Let the degree of both the factor and variable nodes be bounded by a fixed constant. Let $\gamma = \min_{\mathbf{x},i} P(X_i = x_i|\mathcal{X}_{-i} = \mathbf{x}_{-i})$ be fixed. Let $\tilde{P}$ be the probability distribution returned by* Factor-Graph-Parameter-Learn. *Then we have that, for*

$$D_n(P\|\tilde{P}) + D_n(\tilde{P}\|P) \leq \varepsilon$$

to hold with probability at least $1-\delta$, it suffices that we have a certain number of training examples that does not depend on the number of variables in the network.

The following theorem shows a similar result for Bayesian networks, namely that for a fixed bound on the number of parents per node, the sample complexity dependence on the size of the network is $O(1)$.[12]

9. In particular, we first show that the error contribution from any fixed factor is small with high probability. Then—rather than using a Union bound to ensure the error contributions from all factors are small, which would result in a logarithmic dependence of the sample complexity on the number of factors (or variables)—we use Markov's inequality to show that the error contribution of almost all factors is small with high probability. This leaves us to bound the error contribution of the (few) remaining factors, for which the error contribution is not small. By clipping the probability estimates, we can ensure their error contribution is bounded. A very similar reasoning applies to the case of Theorem 9. (See the proofs of Theorems 8 and 9, given in the appendix, for more details.)
10. This solution assumes that γ is known. If not, we can use a clipping threshold as a function of the number of training examples. Such an adaptive clipping procedure was used by Dasgupta (1997) to derive sample complexity bounds for learning fixed structure Bayesian networks.
11. We note that Theorem 8 assumes the maximum number of factors a variable can participate in is fixed (i.e., it cannot grow with the number of variables in the network). As a consequence, the dependence on the number of factors J and the dependence on the number of variables n are equivalent (up to a constant factor).
12. Complete proofs for Theorems 8 and 9 (and all other results in this paper) are given in the appendix of this paper. In the appendix we actually give a much stronger version of Theorem 9, including dependencies of m on ε,δ,k and a graceful degradation result. We note that for non-binary random variables the clipping procedure is a bit more subtle than for binary random variables. In particular, to ensure that the resulting clipped probabilities sum to one, we might have to subtract a small quantity from the highest probability estimate after the clipping. For example, for

Theorem 9 *Let any* $\varepsilon > 0$ *and* $\delta > 0$ *be given. Let any Bayesian network (BN) structure over n variables with at most k parents per variable be given. Let P be a probability distribution that factors over the BN. Let* $\tilde{P}$ *denote the probability distribution obtained by fitting the conditional probability tables (CPT) entries via maximum likelihood and then clipping each CPT entry to the interval* $[\frac{\varepsilon}{8|\mathrm{val(X_j)}|^3}, 1 - \frac{\varepsilon}{8|\mathrm{val(X_j)}|^3}]$. *Then we have that for*

$$D_n(P \| \tilde{P}) \leq \varepsilon$$

to hold with probability at least $1 - \delta$, *it suffices that we have a certain number of training examples that does not depend on the number of variables in the network.*

Theorems 8 and 9 provide theoretical support for the common practice of learning large graphical models from a relatively small number of training examples. More specifically, the number of training examples can be much smaller than the number of parameters when learning large graphical models. In contrast, for many problems in machine learning, the sample complexity grows roughly linearly or at most as some low-order polynomial in the number of parameters (Vapnik, 1998). The difference in sample complexity relates to the discussion of generative versus discriminative training. Indeed our result generalizes and even strengthens the results of Ng and Jordan (2002). They showed a logarithmic dependence on the number of variables for the very specific case of a graphical model with the naive Bayes structure.

4. Structure Learning

The algorithm described in the previous section uses the known network to establish a Markov blanket for each factor. This Markov blanket is then used to estimate the canonical parameters from empirical data. In this section, we show how we can build on this algorithm to perform structure learning, by first identifying (from the data) an approximate Markov blanket for each candidate factor, and then using this approximate Markov blanket to compute the parameters of that factor from a "small" number of training examples.

4.1 Identifying Markov Blankets

In the parameter learning results, the Markov blanket $\mathrm{MB}(\mathbf{C}_j^*)$ is used to efficiently estimate the conditional probability $P(\mathbf{C}_j^* | \mathcal{X} - \mathbf{C}_j^*)$, which is equal to $P(\mathbf{C}_j^* | \mathrm{MB}(\mathbf{C}_j^*))$. This suggests to measure the quality of a candidate Markov blanket $\mathbf{Y}$ by how well $P(\mathbf{C}_j^* | \mathbf{Y})$ approximates $P(\mathbf{C}_j^* | \mathcal{X} - \mathbf{C}_j^*)$. In this section we show how conditional entropy can be used to find a candidate Markov blanket that gives a good approximation for this conditional probability.[13]

ε sufficiently small, we have that naively clipping the probability estimates $(0,0,1/4,3/4)$ to the interval $(\varepsilon, 1-\varepsilon)$ results in $(\varepsilon,\varepsilon,1/4,3/4)$, which does not sum to one (but rather to $1+2\varepsilon$). Subtracting the additional probability mass 2ε from the highest entry fixes this problem. For this example we get $(\varepsilon,\varepsilon,1/4,3/4-2\varepsilon)$. In general, for v-valued random variables, the probability estimates can be made to sum to one (after clipping) by subtracting at most $(v-1)\varepsilon$ from the highest probability estimate. In the appendix we expand more on the topic of clipping for non-binary random variables.

13. For some readers, some intuition might be gained from the fact that the conditional entropy of $\mathbf{C}_j^*$ given the candidate Markov blanket $\mathbf{Y}$ corresponds to the log-loss of predicting $\mathbf{C}_j^*$ given the candidate Markov blanket $\mathbf{Y}$.

Definition 10 (Conditional Entropy) *Let P be a probability distribution over over $\mathbf{X},\mathbf{Y}$. Then the conditional entropy $H(\mathbf{X}|\mathbf{Y})$ of $\mathbf{X}$ given $\mathbf{Y}$ is defined as*

$$-\sum_{\mathbf{x}\in\mathrm{val}(\mathbf{X}),\mathbf{y}\in\mathrm{val}(\mathbf{Y})} P(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y})\log P(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y}).$$

Proposition 11 (Cover & Thomas, 1991) *Let P be a probability distribution over $\mathbf{X},\mathbf{Y},\mathbf{Z}$. Then we have $H(\mathbf{X}|\mathbf{Y},\mathbf{Z})\leq H(\mathbf{X}|\mathbf{Y})$.*

Proposition 11 shows that conditional entropy can be used to find the Markov blanket for a given set of variables. Namely, let $\mathbf{D},\mathbf{Y}\subseteq\mathcal{X}$, $\mathbf{D}\cap\mathbf{Y}=\emptyset$, then we have

$$H(\mathbf{D}|\mathrm{MB}(\mathbf{D}))=H(\mathbf{D}|\mathcal{X}-\mathbf{D})\leq H(\mathbf{D}|\mathbf{Y}), \tag{12}$$

where the equality follows from the Markov blanket property stated in Eqn. (1) and the inequality follows from Proposition 11. Thus, we can select the set of variables $\mathbf{Y}$ that minimizes $H(\mathbf{D}|\mathbf{Y})$ as our candidate Markov blanket for the set of variables $\mathbf{D}$.

Our first difficulty is that, when learning from data, we do not have the true distribution, and hence the exact conditional entropies are unknown. The following lemma shows that the conditional entropy can be efficiently estimated from samples.

Lemma 12 *Let P be a probability distribution over $\mathbf{X},\mathbf{Y}$ such that for all instantiations $\mathbf{x},\mathbf{y}$ we have $P(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y})\geq\lambda$. Let $\widehat{H}$ be the conditional entropy computed based upon m i.i.d. samples from P. Then for*

$$\left|H(\mathbf{X}|\mathbf{Y})-\widehat{H}(\mathbf{X}|\mathbf{Y})\right|\leq\varepsilon$$

to hold with probability $1-\delta$, it suffices that:

$$m\geq\tfrac{8|\mathrm{val}(\mathbf{X})|^2|\mathrm{val}(\mathbf{Y})|^2}{\lambda^2\varepsilon^2}\log\tfrac{4|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|}{\delta}.$$

However, as the empirical estimates of the conditional entropy are noisy, the true Markov blanket is *not* guaranteed to achieve the minimum of $H(\mathbf{D}|\mathbf{Y})$. In fact, in some probability distributions, many sets of variables could be arbitrarily close to reaching equality in Eqn. (12). Thus, in many cases, our procedure will not recover the actual Markov blanket, when given only a finite number of training examples. Fortunately, as we show in the next lemma, any set of variables $\mathbf{U}\cup\mathbf{W}$ that is close to achieving equality in Eqn. (12) gives an accurate approximation $P(\mathbf{C}_j|\mathbf{U},\mathbf{W})$ of the probabilities $P(\mathbf{C}_j|\mathcal{X}-\mathbf{C}_j)$ used in the canonical parameterization.

Lemma 13 *Let any $\varepsilon>0$ be given. Let P be a distribution over disjoint sets of random variables $\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{X},\mathbf{Y}$. Let $\lambda_1=\min_{\mathbf{u}\in\mathrm{val}(\mathbf{U}),\mathbf{v}\in\mathrm{val}(\mathbf{V}),\mathbf{w}\in\mathrm{val}(\mathbf{W})}P(\mathbf{u},\mathbf{v},\mathbf{w})$, and let $\lambda_2=\min_{\mathbf{x}\in\mathrm{val}(\mathbf{X}),\mathbf{u}\in\mathrm{val}(\mathbf{U}),\mathbf{v}\in\mathrm{val}(\mathbf{V}),\mathbf{w}\in\mathrm{val}(\mathbf{W})}P(\mathbf{x}|\mathbf{u},\mathbf{v},\mathbf{w})$. Assume the following holds:*

$$\mathbf{X}\perp\mathbf{Y},\mathbf{W}\ \mid\ \mathbf{U},\mathbf{V}, \tag{13}$$

$$H(\mathbf{X}|\mathbf{U},\mathbf{W})\leq H(\mathbf{X}|\mathbf{U},\mathbf{V},\mathbf{W},\mathbf{Y})+\varepsilon. \tag{14}$$

Then we have that $\forall\ \mathbf{x},\mathbf{y},\mathbf{u},\mathbf{v},\mathbf{w}$

$$\left|\log P(\mathbf{x}|\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y})-\log P(\mathbf{x}|\mathbf{u},\mathbf{w})\right|\leq\tfrac{\sqrt{2\varepsilon}}{\lambda_2\sqrt{\lambda_1}}. \tag{15}$$

In other words, if a set of variables $\mathbf{U} \cup \mathbf{W}$ looks like a Markov blanket for $\mathbf{X}$, as evaluated by the conditional entropy $H(\mathbf{X}|\mathbf{U},\mathbf{W})$, then the conditional distribution $P(\mathbf{X}|\mathbf{U},\mathbf{W})$ must be close to the conditional distribution $P(\mathbf{X}|\mathcal{X}-\mathbf{X})$. Thus, it suffices to find such an approximate Markov blanket $\mathbf{U} \cup \mathbf{W}$ as a substitute for knowing the true Markov blanket $\mathbf{U} \cup \mathbf{V}$. This makes conditional entropy suitable for structure learning.

4.2 Structure Learning Algorithm

We propose the following Factor-Graph-Structure-Learn algorithm. The algorithm receives as input: training examples $\{\mathbf{x}^{(i)}\}_{i=1}^m$; k: the maximum number of variables per factor; b: the maximum number of variables per Markov blanket for any set of variables up to size k; $\bar{\mathbf{x}}$: a base instantiation.[14]

Let $\mathcal{C}$ be the set of candidate factor scopes, let $\mathcal{Y}$ be the set of candidate Markov blankets. I.e., we have

$$\begin{aligned} \mathcal{C} &= \{\mathbf{C}_j^* : \mathbf{C}_j^* \subseteq \mathcal{X}, \mathbf{C}_j^* \neq \emptyset, |\mathbf{C}_j^*| \leq k\}, && (16)\\ \mathcal{Y} &= \{\mathbf{Y} : \mathbf{Y} \subseteq \mathcal{X}, |\mathbf{Y}| \leq b\}. && (17) \end{aligned}$$

The algorithm does the following:

- $\forall\, \mathbf{C}_j^* \in \mathcal{C}$, find $\widehat{\mathrm{MB}}(\mathbf{C}_j^*) = \arg\min_{\mathbf{Y} \in \mathcal{Y}, \mathbf{Y} \cap \mathbf{C}_j^* = \emptyset} \widehat{H}(\mathbf{C}_j^* | \mathbf{Y})$, which is the best candidate Markov blanket.
- $\forall\, \mathbf{C}_j^* \in \mathcal{C}$, compute the estimates $\{\hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}\}_j$ of the canonical factors as defined in Eqn. (7) using the empirical distribution.
- Threshold to one the factor entries $\hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)$ satisfying $|\log \hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)| \leq \frac{\varepsilon}{2^{k+2}}$, and discard the factors that have all entries equal to one.
- Return the probability distribution $\tilde{P}(\mathbf{x}) \propto \prod_j \hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)$.

The thresholding step finds the factors that actually contribute to the distribution. The specific threshold is chosen to suit the proof of Theorem 15. If no thresholding were applied, the error in Eqn. (18) would be $\frac{|\mathcal{C}|}{2^k}\varepsilon$ instead of $J\varepsilon$, which is much larger in case the true distribution has a relatively small number of factors J.

Theorem 14 (Structure learning: computational complexity) *The running time*[15] *of* Factor-Graph-Structure-Learn *is in* $O\big(mkn^kbn^b(k+b) + kn^kbn^bv^{k+b} + kn^k2^kv^k\big)$.

Thus the running time is exponential in the maximum factor scope size k and the maximum Markov blanket size b, polynomial in the number of variables n and the maximum domain size v, and linear in the number of training examples m.

The first two terms in Theorem 14 result from going through the data and computing the empirical conditional entropies. Since the algorithm considers all combinations of candidate factors and Markov blankets, we have an exponential dependence on the maximum scope size k and the

14. Note in the parameter learning setting we had b equal to the size the largest Markov blanket for an *actual* factor in the distribution. In contrast, now b corresponds to the size of the largest Markov blanket for any *candidate* factor up to size k.
15. The upper bound is based on a very naive implementation's running time.

maximum Markov blanket size b. The last term comes from computing the Markov blanket canonical factors. Importantly, unlike for currently-known (exact) ML approaches, the running time does not depend on the tractability of inference in the (unknown) factor graph from which the data was sampled, nor on the tractability of inference in the recovered factor graph.

Theorem 15 (Structure learning: sample complexity) *Let any* $\varepsilon, \delta > 0$ *be given. Let* Factor-Graph-Structure-Learn *be given (a) m training examples* $\{\mathbf{x}^{(i)}\}_{i=1}^m$ *drawn i.i.d. from a distribution P, (b) an upper bound k on the number of variables per factor in the factor graph for P, and (c) an upper bound b on the number of variables per Markov blanket for any set of variables up to size k in the factor graph for P. Let* $\tilde{P}$ *be the distribution returned by* Factor-Graph-Structure-Learn. *Then for*

$$D(P\|\tilde{P}) + D(\tilde{P}\|P) \leq J\varepsilon \tag{18}$$

to hold with probability $1-\delta$, *it suffices that the number of training examples m satisfies:*

$$m \geq \left(1 + \frac{\varepsilon\gamma^{k+b}}{2^{2k+3}}\right)^2 \frac{v^{2k+2b}2^{8k+19}}{\gamma^{6k+6b}\min\{\varepsilon^2,\varepsilon^4\}} \log \frac{8kbn^{k+b}v^{k+b}}{\delta}. \tag{19}$$

Proof *(sketch). From Lemmas 12 and 13 we have that the conditioning set chosen by* Factor-Graph-Structure-Learn *results in a good approximation of the true canonical factor. At this point the structure is fixed, and we can use the sample complexity theorem for parameter learning to finish the proof.* ■

Theorem 15 shows that the sample complexity depends exponentially on the maximum factor size k and the maximum Markov blanket size b; and polynomially on $\frac{1}{\gamma}$ and $\frac{1}{\varepsilon}$. If we modify the analysis to consider the normalized KL-divergence, as in Section 3.5, we obtain a logarithmic dependence on the number of variables in the network.

To understand the implications of this theorem, consider the class of Gibbs distributions where every variable can participate in at most d factors and every factor can have at most k variables in its scope. Then we have that the Markov blanket size $b \leq dk^2$. Bayesian network probability distributions can also be represented using factor graphs.[16] If the number of parents per variable is bounded by numP and the number of children per variable is bounded by numC, then we have $k \leq \text{numP}+1$, and that $b \leq (\text{numC}+1)(\text{numP}+1)^2$. Thus our factor graph structure learning algorithm allows us to efficiently learn distributions that can be represented by Bayesian networks with a bounded number of children and parents per variable. Note that our algorithm recovers a distribution which is close to the true generating distribution, but the distribution it returns is encoded as a factor graph, which may not be representable as a compact Bayesian network.

Theorem 15 considers the case where the generating distribution P factors according to a structure with factor scope sizes bounded by k and size of Markov blankets (of any subset of variables of size less than k) bounded by b. As we did in the case of parameter estimation, we can show that we have graceful degradation of performance for distributions that do not satisfy these assumptions.

16. Given a Bayesian network (BN), the following factor graph represents the same distribution: The factor graph has one variable node per variable in the BN. The factor graph has one factor for each variable in the BN. Each factor's scope is equal to the union of the corresponding variable itself and its parents. Each factor's entries are equal to the corresponding conditional probability table entries of the BN.

Theorem 16 (Structure learning: graceful degradation) *Let any $\varepsilon, \delta > 0$ be given. Let $\{\mathbf{x}^{(i)}\}_{i=1}^m$ be training examples drawn i.i.d. from a distribution Q. Let* MB *and* $\widehat{\mathrm{MB}}$ *be the Markov blankets according to the distributions Q and found by* Factor-Graph-Structure-Learn *respectively. Let $\{f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}\}_j$ be the non-trivial Markov blanket canonical factors of Q (those factors with not all entries equal to one). Let J be the number of non-trivial Markov blanket canonical factors in Q with scope size smaller than k. Let $\tilde{P}$ be the probability distribution returned by* Factor-Graph-Parameter-Learn. *Then we have that for*

$$D(Q\|\tilde{P}) + D(\tilde{P}\|Q) \leq (J+|S|)\varepsilon + 2\sum_{j:|\mathbf{D}^*_j|>k} \max_{\mathbf{d}^*_j} \left|\log f^*_{\mathbf{D}^*_j}(\mathbf{d}_j)\right| + 2\sum_{\mathbf{C}^*_j\in\mathcal{C}\,:\,|\mathrm{MB}(\mathbf{C}^*_j)|>b} \max_{\mathbf{c}^*_j}\left|\log \frac{f^*_{\mathbf{C}^*_j|\mathrm{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)}{f^*_{\mathbf{C}^*_j|\widehat{MB}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)}\right|$$

to hold with probability at least $1-\delta$, it suffices that the number of training examples m satisfies Eqn. (19) of Theorem 15. Here $S = \{j : \mathbf{C}^*_j \notin \{\mathbf{D}_l\}_l, |\mathrm{MB}(\mathbf{C}^*_j)| > b\}$ is the set that indexes over the subsets of variables of size smaller than k over which there is no factor in the true distribution and for which the Markov blanket in the true distribution is larger than b; $\mathcal{C}$ is the set of candidate factor scopes $\mathcal{C} = \{\mathbf{C}^*_j : \mathbf{C}^*_j \subseteq \mathcal{X}, \mathbf{C}^*_j \neq \emptyset, |\mathbf{C}^*_j| \leq k\}$.

Theorem 16 shows that (similar to the parameter learning setting) each canonical factor that is not captured by our learned structure contributes at most a constant to our bound on the KL-divergence (namely $l2^{l+1}\log\frac{1}{\gamma}$ for a factor over l variables, see footnote 8 for details) to our bound on the KL-divergence. This bound on the error contribution can be large, so we discuss the actual error contribution in more detail. The reason a canonical factor is not captured could be two-fold. First, the scope of the factor could be too large. The paragraph after Theorem 7 discusses when the resulting error is expected to be small. Second, the Markov blanket of the factor could be too large. As shown in Lemma 13, a good approximate Markov blanket is sufficient to get a good approximation. So we can expect these error contributions to be small if the true distribution is mostly determined by interactions between small sets of variables.

Recall that the structure learning algorithm correctly clips all estimates of trivial canonical factors to the trivial all-ones factor, when the structural assumptions are satisfied. I.e., trivial factors are correctly estimated as trivial if their Markov blanket is of size smaller than b. The additional term $|S|\varepsilon$ corresponds to estimation error on the factors that are trivial in the true distribution but that have a Markov blanket of size larger than b, and are thus not correctly estimated and clipped to trivial all-ones factors.

5. Related Work

Tables 2 and 3 summarize the prior work on Markov network and Bayesian network learning that comes with formal guarantees. In the following two sections we discuss the prior work on Markov network (factor graph) learning and Bayesian network learning in more detail. We also discuss algorithms that do not have formal guarantees.

Target distribution	True distribution	Structure/Parameter	Samples	Time	Graceful degradation	Reference
ML tree	any	structure	poly	poly	yes	[1]
ML bounded tree-width	any	structure	poly	NP-hard	yes	[2]
Bounded tree-width	same	structure	poly	poly	no	[3]
Factor graph	same	parameter	infinite	convex	no	[4], [5]
Factor graph	same	parameter	poly	poly	yes	[6]
Factor graph	same	structure	poly	poly	yes	[6]

Table 2: Overview of prior work on learning Markov networks that has formal guarantees. More details are given in Section 5.1. The references in the table are: [1]: Chow and Liu (1968); [2] Srebro (2001); [3]: Narasimhan and Bilmes (2004); [4]: Besag (1974b); [5]: Gidas (1988); [6]: this paper. "Convex" refers to the time of solving a convex optimization problem.

5.1 Markov Networks

We split the discussion into two parts: parameter learning and structure learning.

5.1.1 Parameter Learning

The most natural algorithm for parameter estimation in undirected graphical models is maximum likelihood (ML) estimation (possibly with some regularization). Unfortunately, evaluating the likelihood of such a model requires evaluating the partition function. All currently known ML algorithms for undirected graphical models require evaluating the partition function. Therefore, they are computationally tractable only for networks in which inference is computationally tractable. In contrast, our closed form solution can be efficiently computed from the data, even for Markov networks where inference is intractable. Note that our estimator does not return the ML solution, so that our result does not contradict the "hardness" of ML estimation. However, it does provide a low KL-divergence estimate of the probability distribution, with high probability, from a "small" number of training examples, assuming the true distribution approximately factors according to the given structure.

Criteria different from ML have been proposed for learning Markov networks. The most prominent one is *pseudo-likelihood* (Besag, 1974b), and its extension, generalized pseudo-likelihood (Huang and Ogata, 2002). The pseudo-likelihood criterion gives rise to a tractable convex optimization problem. Pseudo-likelihood estimation is consistent, that is, in the infinite sample limit it returns the true distribution, when the assumed structure is correct. (See, for example, Gidas, 1988, .) However, in the finite sample case the pseudo-likelihood estimate is often significantly worse than the maximum likelihood estimate. More information on the statistical efficiency of the pseudo-likelihood estimate can be found in, for example, Besag (1974a); Geyer and Thompson (1992); Guyon and Künsch (1992). In contrast to our results, no finite sample bounds have been provided for pseudo-likelihood estimation. Moreover, the theoretical analyses (e.g., Geman and Graffigne, 1986; Comets, 1992; Guyon and Künsch, 1992) only apply when the generating model is in the true target class.

5.1.2 Structure Learning

Structure learning for Markov networks is notoriously difficult, as it is generally based on using ML estimation of the parameters (with smoothing), often combined with a penalty term for structure complexity. As evaluating the likelihood is only possible for the class of Markov networks in which

Target distribution	True distribution	Structure/Parameter	Samples	Time	Graceful degradation	Reference
ML polytree	any	structure	poly	NP-hard	yes	[1], [2]
ML BN	any	structure	poly	NP-hard	yes	[1], [3]
BN	same	structure	infinite	poly	yes	[4], [5]
Factor graph	BN (same)	structure	poly	poly	yes	[6]

Table 3: Overview of prior work on learning Bayesian networks that has formal guarantees. More details are given in Section 5.2. The references in th table are: [1]: Höffgen (1993); [2]: Dasgupta (1999); [3]: Chickering et al. (2003); [4]: Spirtes et al. (2000); [5]: Cheng et al. (2002); [6]: this paper.

inference is tractable, there have been two main research tracks for ML structure learning. The first, starting with the work of Della Pietra et al. (1997), uses local-search heuristics to add factors into the network (see also McCallum, 2003). The second searches for a structure within a restricted class of models in which inference is tractable, more specifically, bounded tree-width Markov networks. Indeed, ML learning of the class of tree Markov networks—networks of tree-width 1—can be performed very efficiently (Chow and Liu, 1968). Unfortunately, Srebro (2001) proves that for any tree-width k greater than 1, even finding the ML tree-width-k network is NP-hard. Karger and Srebro (2001) provide an approximation algorithm but the approximation factor is a very large multiplicative factor of the log-likelihood. In particular, for tree-width k, they find a Markov network (of tree-width k) with log-likelihood at least $1/(8^k k!(k+1)!)$ times the optimal log-likelihood. Several heuristic algorithms to learn models with small tree-width have been proposed (Malvestuto, 1991; Bach and Jordan, 2002; Deshpande et al., 2001), but (not surprisingly, given the NP-hardness of the problem) they do not come with any performance guarantees.

Recently, Narasimhan and Bilmes (2004) provided a polynomial time algorithm with a polynomial sample complexity guarantee for the class of Markov networks of bounded tree-width. Their algorithm computes approximate conditional independence information followed by dynamic programming to recover the bounded tree-width structure. The parameters for the recovered bounded tree-width model are estimated by standard ML methods. Our algorithm applies to a different family of distributions: factor graphs of bounded connectivity (including graphs in which inference is intractable). Factor graphs with small connectivity can have large tree-width (e.g., grids) and factor graphs with small tree-width can have large connectivity (e.g., star graphs). Thus, the range of applicability is incomparable. Narasimhan and Bilmes (2004) did not provide any graceful degradation guarantees when the generating distribution is not a member of the target class. However, future research might extend their algorithm to this setting.

Pseudo-likelihood has been extended to a criterion for model selection: the resulting criterion is statistically consistent (Ji and Seymour, 1996). In particular they show that the probability of selecting an incorrect model goes to zero as the number of training examples goes to infinity. They also provide a bound on how fast this probability goes to zero. Importantly, Ji and Seymour (1996) only provide a model selection criterion. They do *not* provide an algorithm to efficiently find the best pseudo-likelihood model (according to their evaluation criterion) over the super-exponentially large set of candidate models from which we want to select in the structure learning problem.

5.2 Bayesian Networks

Again, we split the discussion into two parts: parameter learning and structure learning.

5.2.1 PARAMETER LEARNING

ML parameter learning in Bayesian networks (possibly with smoothing) only requires computing the empirical conditional probabilities of each variable given its parent instantiations. Thus there is no computational challenge.

Dasgupta (1997), following earlier work by Friedman and Yakhini (1996), analyzes the sample complexity of learning Bayesian networks, showing that it is polynomial in the maximal number of different instantiations per family. His sample complexity result has logarithmic dependence on the number of variables in the network, when using the KL-divergence normalized by the number of variables in the network. In this paper, we strengthen his result, showing an $O(1)$ dependence of the number of training examples on the number of variables in the network. So for bounded fan-in Bayesian networks, the sample complexity is independent of the number of variables in the network.

5.2.2 STRUCTURE LEARNING

Results analyzing the complexity of structure learning of Bayesian networks fall largely into two classes. The first class of results assumes that the generating distribution is DAG-perfect with respect to some DAG G with at most k parents for each node. (That is, P and G satisfy precisely the same independence assertions.) In this case, algorithms based on various independence tests (Spirtes et al., 2000; Cheng et al., 2002) can identify the correct network structure in the infinite sample limit (i.e., when given an infinite number of training examples), using a polynomial number of independence tests. The infinite sample limit setting is critical in their analysis since it allows for exact independence tests. Neither Spirtes et al. (2000) nor Cheng et al. (2002) provide guarantees for the case of a finite number of training examples, but future research might extend their results to this setting. Chickering and Meek (2002) relax the assumption that the distribution be DAG-perfect; they show that, under a certain assumption, a simple greedy algorithm will, in the infinite sample limit, identify a network structure which is a minimal I-map of the distribution. They provide no polynomial time guarantees, but future work might provide such guarantees for models with bounded connectedness (such as the ones our algorithm considers).

The second class of results relates to the problem of finding a network structure whose score is high, for a given set of training examples and some appropriate scoring function. Although finding the highest-scoring tree-structured network can be done in polynomial time (Chow and Liu, 1968), Chickering (1996) shows that the problem of finding the highest scoring Bayesian network where each variable has at most k parents is NP-hard, for any $k \geq 2$. (See Chickering et al., 2003, for details.) Even finding the maximum likelihood structure among the class of polytrees (Dasgupta, 1999) or paths (Meek, 2001) is NP-hard. These results do not address the question of the number of training examples for which the highest scoring network is guaranteed to be close to the true generating distribution.

Höffgen (1993) analyzes the problem of PAC-learning the structure of Bayesian networks with bounded fan-in, showing that the sample complexity depends only logarithmically on the number of variables in the network (when considering KL-divergence normalized by the number of variables in the network). Höffgen does not provide an efficient learning algorithm (and to date, no efficient learning algorithm is known), stating only that if the optimal network for a given data set can be found (e.g., by exhaustive enumeration), it will be close to optimal with high probability.

In contrast, we provide a polynomial-time learning algorithm with similar performance guarantees for Bayesian networks with bounded fan-in and bounded fan-out. However, we note that our algorithm does not construct a Bayesian network representation, but rather a factor graph; this factor graph may not be compactly representable as a Bayesian network, but it is guaranteed to encode a distribution which is close to the generating distribution, with high probability.

6. Discussion

We have presented the first polynomial-time and polynomial sample-complexity algorithms for both parameter estimation and structure learning in factor graphs of bounded degree. When the generating distribution is within this class of networks, our algorithms are guaranteed to return a distribution close to it, using a polynomial number of training examples. When the generating distribution is not in this class, our algorithm degrades gracefully. Thus our algorithms and analysis are the first to establish the efficient learnability of an important class of distributions.

While of significant theoretical interest, our algorithms, as described, are probably impractical. From a statistical perspective, our algorithm is based on the canonical parameterization, which is evaluated relative to a canonical assignment $\bar{\mathbf{x}}$. Many of the empirical estimates that we compute in the algorithm use only a subset of the training examples that are (in some ways) consistent with $\bar{\mathbf{x}}$. As a consequence, we make very inefficient use of data, in that many training examples may never be used. In regimes where data is not abundant, this limitation may be quite significant in practice. From a computational perspective, our algorithm uses exhaustive enumeration over all possible factors up to some size k, and over all possible Markov blankets up to size b. When we fix k and b to be constant, the complexity is polynomial. But in practice, the set of all subsets of size k or b is often much too large to search exhaustively.

Nevertheless, aside from proving the efficient learnability of an important class of probability distributions, the algorithms we propose might provide insight into the development of new learning algorithms that do work well in practice. In particular, we might be able to address the statistical limitation by putting together canonical factor estimates from multiple canonical assignments $\bar{\mathbf{x}}$. We might be able to address the computational limitation using a more clever (perhaps heuristic) algorithm for searching over subsets. Given the limitations of existing parameter and structure learning algorithms for undirected models, we believe that the techniques suggested by our theoretical analysis are well worth exploring.

Acknowledgments

This work was supported by the Department of the Interior/DARPA under contract number NBCHD030010, and by DARPA's EPCA program, under subcontract to SRI International.

Appendix A. Proofs for Section 2.2

In this section we give formal proofs of all theorems, propositions and lemmas appearing in Section 2.2.

A.1 Proof of Theorem 3

Proof [Theorem 3] The proof consists of two parts:

1. If we let $\{\mathbf{C}_j^*\}_{j=1}^{J^*} = 2^{\mathcal{X}} - \emptyset$, then $P(\mathbf{x}) = P(\bar{\mathbf{x}}) \prod_{j=1}^{J^*} f^*_{\mathbf{C}_j^*}(\mathbf{c}_j^*)$.

2. If P is a positive Gibbs distribution with factor scopes $\{\mathbf{C}_j\}_{j=1}^{J}$, then the canonical factors $f^*_{\mathbf{D}}$ are trivial all-ones factors, whenever $\mathbf{D} \notin \cup_{j=1}^{J} 2^{\mathbf{C}_j}$.

The first part states that the canonical parameterization gives the correct distribution assuming we use a canonical factor for each subset of variables. It is easily verified by counting how often the probabilities $P(\sigma_{\mathbf{U}}[\mathbf{d}])$ contribute for each $\mathbf{U} \subseteq \mathbf{D} \subseteq \mathcal{X}$, and is a standard part of most Hammersley-Clifford theorem proofs. The second part states that we can ignore canonical factors over subsets of variables that do not appear together in one of the factor scopes $\{\mathbf{C}_j\}_{j=1}^{J}$. We now prove the second part. We have

$$\begin{aligned}
\log f^*_{\mathbf{D}}(\mathbf{d}) &= \sum_{\mathbf{U}\subseteq\mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \log P(\sigma_{\mathbf{U}}[\mathbf{d}]) \\
&= \sum_{\mathbf{U}\subseteq\mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \left(\sum_{j=1}^{J} \log f_{\mathbf{C}_j}(\mathbf{C}_j[\sigma_{\mathbf{U}}[\mathbf{d}]]) + \log\frac{1}{Z} \right) \\
&= \sum_{j=1}^{J} \sum_{\mathbf{U}\subseteq\mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \log f_{\mathbf{C}_j}(\mathbf{C}_j[\sigma_{\mathbf{U}}[\mathbf{d}]]). \qquad (20)
\end{aligned}$$

To obtain the last equality, we used the fact that there is an equal number of terms $(\log\frac{1}{Z})$ and $(-\log\frac{1}{Z})$. Now consider the contribution of one factor $f_{\mathbf{C}_j}$ in the above expression. By assumption we have that $\mathbf{D} \notin \cup_{j=1}^{J} 2^{\mathbf{C}_j}$ and thus $\mathbf{D} - \mathbf{C}_j \neq \emptyset$. Now let Y be any element of $\mathbf{D} - \mathbf{C}_j$. Then we have that

$$\begin{aligned}
\sum_{\mathbf{U}\subseteq\mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \log f_{\mathbf{C}_j}(\mathbf{C}_j[\sigma_{\mathbf{U}}[\mathbf{d}]]) = \sum_{\mathbf{U}\subseteq\mathbf{D}-Y} & (-1)^{|\mathbf{D}-\mathbf{U}|} \quad \log \quad f_{\mathbf{C}_j}(\mathbf{C}_j[\sigma_{\mathbf{U}}[\mathbf{d}]]) \\
& + (-1)^{|\mathbf{D}-\mathbf{U}-Y|} \quad \log \quad f_{\mathbf{C}_j}(\mathbf{C}_j[\sigma_{\mathbf{U}\cup Y}[\mathbf{d}]]).
\end{aligned}$$

Now since $Y \notin \mathbf{C}_j$, we have $\mathbf{C}_j[\sigma_{\mathbf{U}}[\mathbf{d}]] = \mathbf{C}_j[\sigma_{\mathbf{U}\cup Y}[\mathbf{d}]]$. And thus we get

$$\sum_{\mathbf{U}\subseteq\mathbf{D}} (-1)^{|\mathbf{D}-\mathbf{U}|} \log f_{\mathbf{C}_j}(\mathbf{C}_j[\sigma_{\mathbf{U}}[\mathbf{d}]]) \quad = \quad 0. \qquad (21)$$

And thus combining Eqn. (21) with Eqn. (20) establishes the second part of the proof. ■

Appendix B. Proofs for Section 3.2

In this section we give formal proofs of all theorems, propositions and lemmas appearing in Section 3.2.

Proof [Proposition 4] In Eqn. (2) the number of terms with a positive sign and a negative sign are both equal to $2^{|\mathbf{D}|-1}$. So we can divide the argument of the log in each term by the same constant $P(\mathcal{X}-\mathbf{D} = (\mathcal{X}-\mathbf{D})[\bar{\mathbf{x}}])$ without changing the factor. The resulting expression is exactly the expression defining $f^*_{\mathbf{D}|\mathcal{X}-\mathbf{D}}$ in Eqn. (7), thus proving the first equality in Eqn. (8). The second equality in Eqn. (8) follows directly from Eqn. (1) and the definition of the factors as functions of probabilities in Eqn. (7). Eqn. (9) and (10) follow directly from Eqn. (8) and Theorem 3. ■

Appendix C. Proofs for Section 3.3

In this section we give formal proofs of all theorems, propositions and lemmas appearing in Section 3.3.

Proof [Theorem 5] The algorithm consists of two parts:

- Collecting the empirical probabilities for each of the factors, jointly with the default instantiation of their Markov blanket. This can be done in three steps. [Below, recall that the maximum factor scope size is k, so there are at most $2^k J$ different canonical factors. Each variable can take on at most v different values.]
  - For all instantiations of all factors initialize the occurrence count to zero. This can be done in $O(2^k J v^k)$.
  - When going through the m data points, we need to add to the counts of the observed instantiation whenever the Markov blanket is in the default instantiation. Reading a specific instantiation of a specific factor and its Markov blanket takes $O(k+b)$ to read every variable. Thus collecting the data counts from which the empirical probabilities will be computed takes $O(m 2^k J(k+b))$.
  - Renormalizing all of the entries to get the empirical conditional probabilities takes time $O(2^k J v^k)$.
- Computing the factor entries from the empirical probabilities. To compute one factor entry $f^*_{\mathbf{C}^*_j}(\mathbf{c}^*_j)$, we have to add (and subtract) $2^{|\mathbf{C}^*_j|}$ empirical log-probabilities. (Note this is the case independent of the cardinality of the variables in the factor, as seen from Eqn. (7).) This gives us $O(2^{|\mathbf{C}^*_j|})$ operations per factor entry, and thus $O(J 2^{2k} v^k)$ total for computing the canonical factor entries from the empirical probabilities.

Adding up the upper bounds on the running times of each step proves the theorem. ■

Appendix D. Proofs for Section 3.4

In this section we give formal proofs of all theorems, propositions and lemmas appearing in Section 3.4.

D.1 Proof of Theorem 6

The proof of the theorem is based on a series of lemmas.

The following lemma shows that the log of the empirical average is an accurate estimate of the log of the population average, if the population average is bounded away from zero.

Lemma 17 *Let any $\varepsilon > 0, \delta > 0, \lambda \in (0,1)$ be given. Let $\{X_i\}_{i=1}^m$ be i.i.d. Bernoulli(ϕ) random variables, where $\lambda \leq \phi \leq 1-\lambda$. Let $\hat{\phi} = \frac{1}{m}\sum_{i=1}^m X_i$. Then for*

$$|\log\phi - \log\hat{\phi}| \leq \varepsilon$$

to hold w.p. $1-\delta$, it suffices that

$$m \geq \frac{(1+\varepsilon)^2}{2\lambda^2\varepsilon^2}\log\frac{2}{\delta}.$$

Proof *From the Hoeffding inequality we have that for*

$$|\phi - \hat{\phi}| \leq \varepsilon'$$

to hold w.p. $1-\delta$ it suffices that

$$m \geq \frac{1}{2\varepsilon'^2}\log\frac{2}{\delta}. \tag{22}$$

Since the function $f(x) = \log x$ is Lipschitz with Lipschitz-constant smaller than $\frac{1}{\lambda-\varepsilon'}$ over the interval $[\lambda-\varepsilon', 1]$, we have that for

$$|\log\phi - \log\hat{\phi}| \leq \frac{\varepsilon'}{\lambda - \varepsilon'}$$

to hold w.p. $1-\delta$, it suffices that m satisfies Eqn. (22). Now for $\frac{\varepsilon'}{\lambda-\varepsilon'} \leq \varepsilon$ to hold, it suffices that $\varepsilon' \leq \frac{\varepsilon\lambda}{1+\varepsilon}$. Using this choice of ε' in Eqn. (22) gives the condition for m as stated in the lemma. ■

The following lemma shows that for distributions that are bounded away from zero, conditional probabilities can be accurately estimated from a small number of samples.

Lemma 18 *Let any $\varepsilon, \delta > 0$ be given. Let $\{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^m$ be i.i.d. samples from a distribution P over $\mathbf{X}, \mathbf{Y}$. Let $\hat{P}$ be the empirical distribution. Let $\lambda = \min_{\mathbf{x},\mathbf{y}} P(\mathbf{X}=\mathbf{x}, \mathbf{Y}=\mathbf{y})$. Then for*

$$|\log P(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y}) - \log\hat{P}(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y})| \leq \varepsilon$$

to hold for all $\mathbf{x}, \mathbf{y}$ with probability $1-\delta$, it suffices that

$$m \geq \frac{(1+\frac{\varepsilon}{2})^2}{2\lambda^2(\frac{\varepsilon}{2})^2}\log\frac{4|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|}{\delta}.$$

Proof *We have (using the definition of conditional probability and the triangle inequality)*

$$\begin{aligned}
&\big|\log P(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y}) - \log\hat{P}(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y})\big| \\
&\quad = \big|\big(\log P(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y}) - \log P(\mathbf{Y}=\mathbf{y})\big) \\
&\qquad -\big(\log\hat{P}(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y}) - \log\hat{P}(\mathbf{Y}=\mathbf{y})\big)\big| \\
&\quad \leq \big|(\log P(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y}) - \log\hat{P}(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y})\big| \\
&\qquad +\big|(\log P(\mathbf{Y}=\mathbf{y}) - \log\hat{P}(\mathbf{Y}=\mathbf{y})\big|.
\end{aligned}$$

Now using Lemma 17 (note that $\lambda = \min_{\mathbf{x},\mathbf{y}} P(\mathbf{X}=\mathbf{x},\mathbf{Y}=\mathbf{y}) \leq \min_{\mathbf{y}} P(\mathbf{Y}=\mathbf{y})$*) and the Union bound to bound both terms by* $\varepsilon/2$*, we get that for*

$$\left|\log P(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y}) - \log \hat{P}(\mathbf{X}=\mathbf{x}|\mathbf{Y}=\mathbf{y})\right| \leq \varepsilon \tag{23}$$

to hold with probability $1-2\delta'$*, it is sufficient that*

$$m \geq \frac{(1+\varepsilon/2)^2}{2\lambda^2(\varepsilon/2)^2}\log\frac{2}{\delta'}. \tag{24}$$

Using the Union bound, we get that for Eqn. (23) to hold with probability $1-2|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|\delta'$ *for all* $\mathbf{x}\in\mathrm{val}(\mathbf{X}),\mathbf{y}\in\mathrm{val}(\mathbf{Y})$ *it suffices that m satisfies Eqn. (24). Choosing* $\delta = 2|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|\delta'$ *gives the statement of the lemma.* ∎

Our algorithm uses probability estimates to compute canonical factors. The following lemma shows that accurate probabilities are sufficient to obtain accurate canonical factors.

Lemma 19 *Let any* $\varepsilon > 0$ *be given. Let any* $\mathbf{D},\mathbf{Y},\mathbf{W} \subseteq \mathcal{X}, \mathbf{D}\cap\mathbf{Y}=\emptyset, \mathbf{D}\cap\mathbf{W}=\emptyset$ *be given. Then for all* $\mathbf{d}\in\mathrm{val}(\mathbf{D})$ *for*

$$|\log f^*_{\mathbf{D}|\mathbf{Y}}(\mathbf{d}) - \log \hat{f}^*_{\mathbf{D}|\mathbf{W}}(\mathbf{d})| \leq \varepsilon$$

to hold, it suffices that for all instantiations $\mathbf{d}\in\mathrm{val}(\mathbf{D})$ *we have that*

$$|\log P(\mathbf{d}|\bar{\mathbf{y}}) - \log \hat{P}(\mathbf{d}|\bar{\mathbf{w}})| \leq \tfrac{\varepsilon}{2^{|\mathbf{D}|}}. \tag{25}$$

Proof

$$\begin{aligned}
\left|\log f^*_{\mathbf{D}|\mathbf{Y}}(\mathbf{d}) - \log \hat{f}^*_{\mathbf{D}|\mathbf{W}}(\mathbf{d})\right| &= \Big|\sum_{\mathbf{Z}\subseteq\mathbf{D}}(-1)^{|\mathbf{D}-\mathbf{Z}|}\log P(\sigma_{\mathbf{Z}:\mathbf{D}}[\mathbf{d}]|\mathbf{Y}=\bar{\mathbf{y}}) \\
&\quad -\sum_{\mathbf{Z}\subseteq\mathbf{D}}(-1)^{|\mathbf{D}-\mathbf{Z}|}\log \hat{P}(\sigma_{\mathbf{Z}:\mathbf{D}}[\mathbf{d}]|\mathbf{W}=\bar{\mathbf{w}})\Big| \\
&\leq \sum_{\mathbf{Z}\subseteq\mathbf{D}}\Big|\log P(\sigma_{\mathbf{Z}:\mathbf{D}}[\mathbf{d}]|\mathbf{Y}=\bar{\mathbf{y}}) \\
&\quad -\log \hat{P}(\sigma_{\mathbf{Z}:\mathbf{D}}[\mathbf{d}]|\mathbf{W}=\bar{\mathbf{w}})\Big| \\
&\leq \sum_{\mathbf{Z}\subseteq\mathbf{D}}\frac{\varepsilon}{2^{|\mathbf{D}|}} \\
&= \varepsilon,
\end{aligned}$$

where, in order, we used the definitions of f^* *and* $\hat{f}^*$*; triangle inequality; Eqn. (25); number of subsets of* $\mathbf{D}$ *equals* $2^{|\mathbf{D}|}$. ∎

The next step is to show that, if we obtain good estimates of the factors, the distributions they induce should be close as well. The following lemma shows that distributions with approximately the same factors are close to each other, by proving a bound on $D(P\|\hat{P}) + D(\hat{P}\|P)$, and thus (since $D(\cdot\|\cdot)\geq 0$) a bound on $D(P\|\hat{P})$.

Lemma 20 *Let* $P(\mathbf{x}) = \frac{1}{Z}\prod_{j=1}^{J} f_j(\mathbf{c}_j)$ *and* $\hat{P}(\mathbf{x}) = \frac{1}{\hat{Z}}\prod_{j=1}^{J} \hat{f}_j(\mathbf{c}_j)$. *Let* $\varepsilon = \max_{j\in\{1,\cdots,J\},\mathbf{c}_j} |\log f_j(\mathbf{c}_j) - \log \hat{f}_j(\mathbf{c}_j)|$. *Then we have that*

$$D(P\|\hat{P}) + D(\hat{P}\|P) \leq 2J\varepsilon.$$

Proof

$$\begin{aligned} D(P\|\hat{P}) + D(\hat{P}\|P) &= \mathrm{E}_{\mathbf{X}\sim P}\left(\log P(\mathbf{X}) - \log \hat{P}(\mathbf{X})\right) + \mathrm{E}_{\mathbf{X}\sim \hat{P}}\left(\log \hat{P}(\mathbf{X}) - \log P(\mathbf{X})\right) \\ &= \mathrm{E}_{\mathbf{X}\sim P}\sum_{j=1}^{J}\left(\log f_j(\mathbf{C}_j^*) - \log \hat{f}_j(\mathbf{C}_j^*)\right) - \log\frac{Z}{\hat{Z}} \\ &\quad + \mathrm{E}_{\mathbf{X}\sim \hat{P}}\sum_{j=1}^{J}\left(\log \hat{f}_j(\mathbf{C}_j^*) - \log f_j(\mathbf{C}_j^*)\right) - \log\frac{\hat{Z}}{Z} \\ &\leq 2J\varepsilon, \end{aligned}$$

where we used in order: the definition of KL-divergence; the definition of P, $\hat{P}$; $\log\frac{Z}{\hat{Z}} + \log\frac{\hat{Z}}{Z} = 0$, *and the fact that each term in the expectation is bounded in absolute value by* ε. ■

Note that (by using the sum of the KL-divergences) we have that the terms that involve the partition functions Z and $\hat{Z}$ cancel. This enables us to prove an error bound without bounding the difference $|\log Z - \log \hat{Z}|$ as a function of the errors in the factors.

We now show how the previous lemmas can be used to prove the parameter learning sample complexity result stated in Theorem 6.

Proof [Theorem 6] First note that since the scopes of the canonical factors used by the algorithm are subsets of the given scopes $\{\mathbf{C}_j\}_{j=1}^{J}$, we have that

$$\max_j |\mathbf{C}_j^* \cup \mathrm{MB}(\mathbf{C}_j^*)| \quad \leq \quad b+k.$$

Let $\hat{P}$ be the empirical distribution as given by the samples $\{\mathbf{x}^{(i)}\}_{i=1}^{m}$. Let $\mathbf{M}_j^* = \mathrm{MB}(\mathbf{C}_j^*)$. Then from Lemma 18 we have that for any $j \in \{1,\ldots,J^*\}$ for

$$\left|\log P(\mathbf{C}_j^* = \mathbf{c}_j^* | \mathbf{M}_j^* = \mathbf{m}_j^*) - \log \hat{P}(\mathbf{C}_j^* = \mathbf{c}_j^* | \mathbf{M}_j^* = \mathbf{m}_j^*)\right| \leq \varepsilon' \tag{26}$$

to hold for all instantiations $\mathbf{c}_j^*, \mathbf{m}_j^*$ with probability $1-\delta'$, it suffices that

$$m \geq \frac{(1+\frac{\varepsilon'}{2})^2}{2\gamma^{2k+2b}(\frac{\varepsilon'}{2})^2}\log\frac{4v^{k+b}}{\delta'}. \tag{27}$$

Using Lemma 19 we obtain that for all instantiations $\mathbf{c}_j^*$ we have that Eqn. (26) implies

$$|\log f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) - \log \hat{f}^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)| \leq 2^k\varepsilon'. \tag{28}$$

Using the union bound, we get that for Eqn. (28) to hold for all $j \in J^*$ with probability $1 - J^*\delta'$, it suffices that m satisfies Eqn. (27). When Eqn. (28) holds for all $j \in J^*$, Lemma 20 and Proposition 4 give us that

$$D(P\|\tilde{P}) + D(\tilde{P}\|P) \leq 2J^* 2^k \varepsilon'. \tag{29}$$

We have that $J^* \leq 2^k J$. Choosing $\varepsilon' = \frac{\varepsilon}{2^{2k+1}}$ and $\delta' = \frac{\delta}{2^k J}$ and substituting these choices into Eqn. (27) and Eqn. (29) gives the theorem. ■

D.2 Proof of Theorem 7

Proof [Theorem 7] From Proposition 4 we have that

$$Q(\mathbf{x}) = \frac{1}{Z}\prod_{j=1}^{\bar{J}} f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j).$$

We can rewrite this product as follows:

$$\begin{aligned} Q(\mathbf{x}) &= \frac{1}{Z} \prod_{j:\ \substack{\mathbf{D}^*_j \in \{\mathbf{C}^*_k\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}^*_j) = \widehat{MB}(\mathbf{D}^*_j)}} f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j) \\ &\quad \prod_{j:\ \substack{\mathbf{D}^*_j \in \{\mathbf{C}^*_k\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}^*_j) \neq \widehat{MB}(\mathbf{D}^*_j)}} f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j) \\ &\quad \prod_{j:\mathbf{D}^*_j \notin \{\mathbf{C}^*_k\}_{k=1}^{J^*}} f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j). \end{aligned} \tag{30}$$

We also have

$$\tilde{P}(\mathbf{x}) = \frac{1}{\tilde{Z}}\prod_{j=1}^{J^*} \hat{f}^*_{\mathbf{C}^*_j|\widehat{\mathrm{MB}}(\mathbf{C}^*_j)}(\mathbf{c}^*_j). \tag{31}$$

We can rewrite this product as follows:

$$\begin{aligned} \tilde{P}(\mathbf{x}) &= \frac{1}{\tilde{Z}} \prod_{j:\ \substack{\mathbf{D}^*_j \in \{\mathbf{C}^*_k\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}^*_j) = \widehat{MB}(\mathbf{D}^*_j)}} \hat{f}^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j) \\ &\quad \prod_{j:\ \substack{\mathbf{D}^*_j \in \{\mathbf{C}^*_k\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}^*_j) \neq \widehat{MB}(\mathbf{D}^*_j)}} \hat{f}^*_{\mathbf{D}^*_j|\widehat{\mathrm{MB}}(\mathbf{D}^*_j)}(\mathbf{d}^*_j) \\ &\quad \prod_{j:\ \substack{\mathbf{C}^*_j \notin \{\mathbf{D}^*_k\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}^*_j) = \widehat{MB}(\mathbf{C}^*_j)}} \hat{f}^*_{\mathbf{C}^*_j|\widehat{\mathrm{MB}}(\mathbf{C}^*_j)}(\mathbf{c}^*_j) \\ &\quad \prod_{j:\ \substack{\mathbf{C}^*_j \notin \{\mathbf{D}^*_k\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}^*_j) \neq \widehat{MB}(\mathbf{C}^*_j)}} \hat{f}^*_{\mathbf{C}^*_j|\widehat{\mathrm{MB}}(\mathbf{C}^*_j)}(\mathbf{c}^*_j). \end{aligned} \tag{32}$$

We have (adding and subtracting same term):

$$\log \frac{f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j)}{\hat{f}^*_{\mathbf{D}^*_j|\widehat{\mathrm{MB}}(\mathbf{D}^*_j)}(\mathbf{d}^*_j)} = \log \frac{f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j)}{f^*_{\mathbf{D}^*_j|\widehat{\mathrm{MB}}(\mathbf{D}^*_j)}(\mathbf{d}^*_j)} + \log \frac{f^*_{\mathbf{D}^*_j|\widehat{\mathrm{MB}}(\mathbf{D}^*_j)}(\mathbf{d}^*_j)}{\hat{f}^*_{\mathbf{D}^*_j|\widehat{\mathrm{MB}}(\mathbf{D}^*_j)}(\mathbf{d}^*_j)}. \tag{33}$$

We also have for $j : \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_k$ that $\log f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{C}_j^*) = 0$. Thus we have (adding zero and adding and subtracting same term):

$$\log \hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) = \log \frac{\hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)} + \log \frac{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}. \tag{34}$$

Using Eqn. (30), Eqn. (32), Eqn. (33) and Eqn. (34) we get that $D(Q\|\tilde{P}) + D(\tilde{P}\|Q) =$

$$E_{\mathbf{X}\sim Q}\Big(\sum_{\substack{j:\ \mathbf{D}_j^* \in \{\mathbf{C}_k^*\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}_j^*) = \widehat{MB}(\mathbf{D}_j^*)}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) - E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{\substack{j:\ \mathbf{D}_j^* \in \{\mathbf{C}_k^*\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}_j^*) = \widehat{MB}(\mathbf{D}_j^*)}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) \tag{35}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{\substack{j:\ \mathbf{D}_j^* \in \{\mathbf{C}_k^*\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}_j^*) \neq \widehat{MB}(\mathbf{D}_j^*)}} \log \frac{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) - E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{\substack{j:\ \mathbf{D}_j^* \in \{\mathbf{C}_k^*\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}_j^*) \neq \widehat{MB}(\mathbf{D}_j^*)}} \log \frac{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) \tag{36}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{\substack{j:\ \mathbf{D}_j^* \in \{\mathbf{C}_k^*\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}_j^*) \neq \widehat{MB}(\mathbf{D}_j^*)}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) - E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{\substack{j:\ \mathbf{D}_j^* \in \{\mathbf{C}_k^*\}_{k=1}^{J^*} \\ \mathrm{MB}(\mathbf{D}_j^*) \neq \widehat{MB}(\mathbf{D}_j^*)}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) \tag{37}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{j:\mathbf{D}_j \notin \{\mathbf{C}_k^*\}_{k=1}^{J^*}} \log f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)\Big) - E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{j:\mathbf{D}_j \notin \{\mathbf{C}_k^*\}_{k=1}^{J^*}} \log f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)\Big) \tag{38}$$

$$-E_{\mathbf{X}\sim Q}\Big(\sum_{\substack{j:\ \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}_j^*) = \widehat{MB}(\mathbf{C}_j^*)}} \log \hat{f}^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)\Big) + E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{\substack{j:\ \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}_j^*) = \widehat{MB}(\mathbf{C}_j^*)}} \log \hat{f}^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)\Big) \tag{39}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{\substack{j:\ \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}_j^*) \neq \widehat{MB}(\mathbf{C}_j^*)}} \log \frac{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{\hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) - E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{\substack{j:\ \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}_j^*) \neq \widehat{MB}(\mathbf{C}_j^*)}} \log \frac{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{\hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) \tag{40}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{\substack{j:\ \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}_j^*) \neq \widehat{MB}(\mathbf{C}_j^*)}} \log \frac{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) - E_{\mathbf{X}\sim \tilde{P}}\Big(\sum_{\substack{j:\ \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}} \\ \mathrm{MB}(\mathbf{C}_j^*) \neq \widehat{MB}(\mathbf{C}_j^*)}} \log \frac{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) \tag{41}$$

$$+\log \frac{\tilde{Z}}{Z} + \log \frac{Z}{\tilde{Z}}. \tag{42}$$

Recall $T = \{j : \mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_{k=1}^{\bar{J}}, \mathrm{MB}(\mathbf{C}_j^*) \neq \widehat{MB}(\mathbf{C}_j^*)\}$. Using the same reasoning as in the proof of Theorem 6, we have that for the sum of the terms in lines (35), (36), (39) and (40) to be bounded by $J\varepsilon$ with probability at least $1-\delta$, it suffices that m satisfies the condition on m in Eqn. (11) of Theorem 6 .

The sum of the terms in lines (37) and (41) can be bounded by

$$2 \sum_{j\,:\,\mathrm{MB}(\mathbf{C}_j^*) \neq \widehat{MB}(\mathbf{C}_j^*)} \max_{\mathbf{c}_j^*} \Big| \log \frac{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)} \Big|.$$

The sum of the terms in lines (38) can be bounded by

$$2 \sum_{j:\mathbf{D}_j^* \notin \{\mathbf{C}_k^*\}_{k=1}^{J^*}} \max_{\mathbf{d}_j^*} \left| \log f^*_{\mathbf{D}_j^*}(\mathbf{d}_j) \right|.$$

The two terms in line (42) sum to zero.

This establishes the theorem. ■

Appendix E. Proofs for Section 3.5

We will treat the proofs for the factor graph case and the Bayesian network case in two separate sections.

E.1 Proof of Theorem 8

Proof [Theorem 8] Using the same reasoning as in the proof of Theorem 6, we get that for any fixed $j \in \{1, \cdots, J^*\}$ for

$$|\log f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) - \log \hat{f}^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)| \leq \frac{\varepsilon'}{2^{k+1}} \tag{43}$$

to hold for all instantiations $\mathbf{c}_j^*$ with probability $1 - \delta'$ it suffices that

$$m \geq \frac{(1 + \frac{\varepsilon'}{2^{2k+2}})^2}{2\gamma^{2k+2b}(\frac{\varepsilon'}{2^{2k+2}})^2} \log \frac{4v^{k+b}}{\delta'}. \tag{44}$$

Also, using the same reasoning as in the proof of Lemma 20, we get that

$$D_n(P\|\tilde{P}) + D_n(\tilde{P}\|P) \leq \frac{2}{n} \sum_{j=1}^{J^*} \max_{\mathbf{c}_j^*} |\log f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) - \log \hat{f}^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)|.$$

We have for all factors and instantiations that (recall that 2^k probabilities contribute to each factor, and each probability is over k variables, thus each (log) conditional probability has maximal skew $\log \frac{(1-\gamma)^k}{\gamma^k} \leq k \log \frac{1}{\gamma}$)

$$|\log f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) - \log \hat{f}^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)| \leq k 2^k \log \frac{1}{\gamma}.$$

Note that clipping of the probability estimates ensures this holds with probability one. Thus we get that

$$\begin{aligned}
\mathrm{E}\big(D_n(P\|\tilde{P}) + D_n(\tilde{P}\|P)\big) &\leq \frac{2}{n} J^* \frac{\varepsilon'}{2^{k+1}} + \delta' \frac{2}{n} J^* k 2^k \log \frac{1}{\gamma} \\
&\leq \frac{J}{n}\varepsilon' + \frac{J}{n} k 2^{2k+1} \delta' \log \frac{1}{\gamma},
\end{aligned}$$

where for the last inequality we used $J^* \leq 2^k J$. The Markov inequality $(P(X \leq \alpha) \geq 1 - \frac{\mathrm{E}X}{\alpha})$ gives us that

$$D_n(P\|\tilde{P}) + D_n(\tilde{P}\|P) \leq \varepsilon \tag{45}$$

holds with probability

$$1 - \frac{\frac{J}{n}\varepsilon' + \frac{J}{n}k2^{2k+1}\delta' \log\frac{1}{\gamma}}{\varepsilon}.$$

Now choosing ε', δ' such that $\frac{\delta}{2} = \frac{J}{n}\frac{\varepsilon'}{\varepsilon} = \frac{J}{n}\frac{k2^{2k+1}\delta' \log\frac{1}{\gamma}}{\varepsilon}$ and substituting this back into the sufficient condition on m, gives us that for Eqn. (45) to hold with probability $1-\delta$, it suffices that

$$m \geq \frac{(1+\frac{n}{J}\frac{\varepsilon\delta}{2^{2k+3}})^2}{2\gamma^{2k+2b}(\frac{n}{J}\frac{\varepsilon\delta}{2^{2k+3}})^2} \log \frac{k2^{2k+4}v^{k+b}\log\frac{1}{\gamma}}{\frac{n}{J}\varepsilon\delta}.$$

Since the number of factors per variables is bounded by a constant, we have that $\frac{J}{n}$ is bounded by that constant. And thus we have that m is $O(1)$ when considering only the dependence on n, the number of variables. ■

E.2 Proof of Theorem 9

For clarity of the overall proof structure of Theorem 9, we defer the proofs of the helper lemmas to the next section. Note the theorem stated in this section is stronger than Theorem 9: it includes dependencies of m on ε, δ, the maximum domain size of the variables v, and the maximum number of parents k. It also shows the graceful degradation for the case of learning a distribution that does not factor according to the given structure.

For any $\gamma < \frac{1}{v}$, and any multinomial distribution with means $\theta_{1:v}$, the multinomial distribution with means clipped to $[\gamma, 1-\gamma]$ refers to the distribution obtained by clipping every θ_i to $[\gamma, 1-\gamma]$, after which the θ_i are adjusted to sum to one, while kept in the interval $[\gamma, 1-\gamma]$. It is easily verified this is always possible without changing any θ_i by more than $v\gamma$. (Although the adjustment such that the entries sum to one need not be unique, it does not matter for our results which choice is made.) We write $D(\theta^{(1)}_{1:v} \| \theta^{(2)}_{1:v})$ as a shortcut for $D(P_1 \| P_2)$, where P_1, P_2 are multinomial distributions with means $\theta^{(1)}_1, \ldots, \theta^{(1)}_v$ and $\theta^{(2)}_1, \ldots, \theta^{(2)}_v$ respectively. The following lemmas establish the basic results used to prove our main sample complexity bounds for Bayesian networks parameter learning.

Lemma 21 *Let any $\delta > 0, \varepsilon > 0$ be fixed, and let there be m i.i.d. samples drawn from a v-valued multinomial distribution with means $\theta^*_{1:v}$, and let $\tilde{\theta}_{1:v}$ be the empirical distribution, clipped to the interval $[\frac{\varepsilon}{4v^3}, 1-\frac{\varepsilon}{4v^3}]$. Then if $m \geq \frac{8v^4}{\varepsilon^2}\log\frac{2v}{\delta}$, we have that $D(\theta^*_{1:v} \| \tilde{\theta}_{1:v}) \leq \varepsilon$ w.p. $1-\delta$.*

Lemma 22 *Let two v-valued multinomial distributions with means $\theta^{(1)}_{1:v} \in [0,1]^v, \theta^{(2)}_{1:v} \in [\gamma, 1-\gamma]^v$ be given. Then we have that $D(\theta^{(1)}_{1:v} \| \theta^{(2)}_{1:v}) \leq \log\frac{1}{\gamma}$.*

Lemma 23 *Let m_H be the sum of m i.i.d. Bernoulli(p) random variables. If $m \geq \frac{8}{p}\log\frac{1}{\delta}$, then we have that $m_H \geq \frac{mp}{2}$ with probability $1-\delta$.*

Lemma 24 *Let $\{X_i\}_{i=1}^{k+1}$ be a set of $k+1$ random variables with $|\mathrm{val}(X_i)| \leq v$ for all $i = 1:k+1$. Let $\mathbf{u} \in \mathrm{val}(X_{1:k})$. Let any $\varepsilon > 0, \delta > 0$ be given. Let $\tilde{P}(X_{k+1}|X_{1:k} = \mathbf{u})$ be the empirical estimate of $X_{k+1}|X_{1:k} = \mathbf{u}$ (based on m independent samples of $\{X_i\}_{i=1}^{k+1}$ drawn from $P(X_{1:k+1})$) clipped to the interval $[\frac{\varepsilon}{4|\mathrm{val}(X_{k+1})|^3}, 1-\frac{\varepsilon}{4|\mathrm{val}(X_{k+1})|^3}]$. Then to ensure that $D(P(X_{k+1}|X_{1:k} = \mathbf{u}) \| \tilde{P}(X_{k+1}|X_{1:k} = \mathbf{u})) \leq \frac{\varepsilon}{v^{k/2}\sqrt{P(X_{1:k}=\mathbf{u})}}$ w.p. $1-\delta$, it suffices that $m \geq \frac{16\, v^{4+k}}{\varepsilon^2}\log^2\frac{4v^3}{\varepsilon}\log\frac{4v}{\delta}$.*

Lemma 25 *Let $\{X_i\}_{i=1}^{k+1}$ be a set of $k+1$ random variables with* $\mathrm{val}(\mathrm{X_i}) \le v$ *for all $i = 1 : k+1$. Let any $\varepsilon > 0, \delta > 0$ be given. For all $\mathbf{u} \in \mathrm{val}(\mathrm{X_{1:k}})$ let $\tilde{P}(X_{k+1}|X_{1:k} = \mathbf{u})$ be the empirical estimate of $X_{k+1}|X_{1:k} = \mathbf{u}$ (based on m independent samples of $\{X_i\}_{i=1}^{k+1}$ drawn from $P(X_{1:k+1})$) clipped to the interval $[\frac{\varepsilon}{4|\mathrm{val}(\mathrm{X_{k+1}})|^3}, 1 - \frac{\varepsilon}{4|\mathrm{val}(\mathrm{X_{k+1}})|^3}]$. Then for $\sum_{\mathbf{u}\in\mathrm{val}(\mathrm{X_{1:k}})} P(X_{1:k} = \mathbf{u}) D(P(X_{k+1}|X_{1:k} = \mathbf{u}) \| \tilde{P}(X_{k+1}|X_{1:k} = \mathbf{u})) \le \varepsilon$ to hold with probability $1-\delta$, it suffices that $m \ge \frac{16\, v^{4+k}}{\varepsilon^2} \log^2 \frac{4v^3}{\varepsilon} \log \frac{4v^{k+1}}{\delta}$.*

Because KL divergence can be unbounded, typically some process, such as clipping, is needed to ensure that our algorithms do not suffer infinite loss. Lemmas 21 and 22 show that we can bound the KL divergence by clipping the (estimated) probabilities away from $\{0,1\}$. (Abe et al. (1991) and Abe et al. (1992) give a more detailed treatment of uniform convergence for KL divergence loss.) Lemma 24 shows how to bound our error on individual conditional probability table (CPT) entries. Note that in Lemma 24, the loss is allowed to be larger for less likely instantiations of the conditioning variables. Also note that Lemma 24 shows that the number of samples m required does not depend on the probability of the instantiations of the conditioning variables, no matter how likely/unlikely. Lemma 23 is used in our proof of Lemma 24 to relate the required number of samples with a specific instantiation $\mathbf{u}$ of the conditioning variables to the actual number of training examples required. Lemma 25 relates the loss on individual CPT entries to the conditional KL divergence, and follows directly from Lemma 24 and Cauchy-Schwarz.

Using the lemmas above, we are now ready to prove a bound on the sample complexity of learning a fixed structure BN. We note that Dasgupta (1997) showed a bound on the sample complexity of BN learning that was polynomial in the number of variables n. His proof method relied on using a Union bound to show that *all* of the n nodes in the BN will have accurate CPT entries, which meant the bound necessarily had to have a dependence on n (even if the normalized KL criterion had been used). For the normalized KL criterion, his method gives a logarithmic dependence on n. Below, we will derive a strictly stronger bound, which has no dependence on the number of variables in the BN. Our bound is based on showing that (i) Given any fixed node, with high probability, its CPT entries will be accurate (Lemma 25), and (ii) Using the Markov inequality to show that, as a consequence, *almost all* of the nodes in the network will have CPT entries that are accurate. This turns out to be sufficient to ensure the estimated BN parameters will provide a good approximation to the joint distribution, and eliminates the bound's dependence on n.

In the theorem below, P is some "true" underlying distribution from which the samples are drawn; P_{BN} is the best possible approximation to P using a given BN structure (in the sense of minimizing $D_n(P\|\cdot)$), and $\tilde{P}_{BN}$ is the learned estimate of P. We give a bound on the number of training examples required for $\tilde{P}_{BN}$'s performance to approach that of P_{BN}.

Theorem 26 *Let any $\varepsilon > 0$ and $\delta > 0$ be fixed. Let P be any probability distribution over n multinomial random variables $X_{1:n}$, where each of the random variables X_i can take on at most v values. Let any BN structure be given, and let k be the maximum number of parents per variable. (P may not factor according to the BN structure.) Let P_{BN} be the best possible estimate of P using a model that factorizes according to the BN structure. (I.e., P_{BN}'s conditional probability distributions satisfy $P_{BN}(X_i|PaX_i) = P(X_i|PaX_i)$.) Let $\tilde{P}_{BN}$ denote the probability distribution obtained by fitting (via maximum likelihood) a BN model with the given structure to the m i.i.d. training examples drawn from P, and then clipping for each X_j each CPT entry to the interval $[\frac{\varepsilon}{8|\mathrm{val}(\mathrm{X_j})|^3}, 1 - \frac{\varepsilon}{8|\mathrm{val}(\mathrm{X_j})|^3}]$. Then, to ensure that with probability $1-\delta$, $\tilde{P}_{BN}$ is nearly as good an estimate as P_{BN} of the true distribution*

P, that is, we have

$$D_n(P\|\tilde{P}_{BN}) \leq D_n(P\|P_{BN}) + \varepsilon,$$

it suffices that the training set size be

$$m \geq 64\frac{v^{4+k}\log^2\frac{8v^3}{\varepsilon}}{\varepsilon^2}\log(\frac{8v^{k+1}}{\varepsilon\delta}\log\frac{8v^3}{\varepsilon}).$$

Remark. *Note that if P does factor according to the given BN structure, then the term $D_n(P\|P_{BN})$ above equals zero.*

Proof *The following equality is easily verified:*

$$\begin{aligned} D(P\|\tilde{P}_{BN}) &= D(P\|P_{BN}) \\ &\quad + \sum_{j=1}^{n}\sum_{u\in\mathrm{val(PaX_j)}} P(PaX_j=\mathbf{u})D(P(X_j|PaX_j=\mathbf{u})\|\tilde{P}(X_j|PaX_j=\mathbf{u})). \end{aligned} \tag{46}$$

From Lemma 25 we have that for estimates clipped to $[\frac{\varepsilon'}{4|\mathrm{val(X_j)}|^3}, 1-\frac{\varepsilon'}{4|\mathrm{val(X_j)}|^3}]$, that for

$$\sum_{\mathbf{u}\in\mathrm{val(PaX_j)}} P(PaX_j=\mathbf{u})D(P(X_j|PaX_j=\mathbf{u})\|\tilde{P}(X_j|PaX_j=\mathbf{u})) \leq \varepsilon'$$

to hold with probability $1-\tau$, it suffices that

$$m \geq 16\frac{v^{4+k}\log^2\frac{4v^3}{\varepsilon'}}{\varepsilon'^2}\log\frac{4v^{k+1}}{\tau}. \tag{47}$$

Now let $Z=\sum_i\eta_i$ be the sum over indicator variables $\eta_i = \mathbf{1}\{\sum_{\mathbf{u}}P(PaX_i=\mathbf{u})D(P(X_j|PaX_j=\mathbf{u})\|\tilde{P}(X_j|PaX_j=\mathbf{u})) > \varepsilon'\}$, and let τ be as above. Then applying the Markov inequality to the non-negative random variable Z gives

$$P(\textstyle\sum_{i=1}^n\eta_i \leq \frac{n\tau}{\delta}) \geq 1-\delta. \tag{48}$$

So, we have that

$$\begin{aligned} D_n(P\|\tilde{P}_{BN}) &\leq D_n(P\|P_{BN}) + \tfrac{1}{n}\textstyle\sum_{i=1}^n(1-\eta_i)\varepsilon' + \tfrac{1}{n}\sum_{i=1}^n\eta_i\log\frac{4v^3}{\varepsilon'} \\ &\leq D_n(P\|P_{BN}) + \varepsilon' + \tfrac{\tau}{\delta}\log\tfrac{4v^3}{\varepsilon'} \quad \textit{w.p. } 1-\delta. \end{aligned} \tag{49}$$

For the first inequality we used Eqn. (46), the definition of η_i and Lemma 22. The second inequality follows from Eqn. (48). To bound the right hand side of Eqn. (49), we bound each of the terms by $\frac{\varepsilon}{2}$. For the first term this implies $\varepsilon'=\frac{\varepsilon}{2}$, for the second term, this allows us to solve for the free parameter $\tau=\frac{\varepsilon\delta}{2\log\frac{4v^3}{\varepsilon'}}=\frac{\varepsilon\delta}{2\log\frac{8v^3}{\varepsilon}}$. Substituting these expressions for ε' and τ into Eqn. (47, 49), gives the statement of the theorem. ■

Note that Eqn. (46) holds for all distributions $\tilde{P}_{BN}$ that factor according to the BN. Since KL divergence is always non-negative, Eqn. (46) implies that $D(P\|P_{BN}) \leq D(P\|\tilde{P}_{BN})$. So the clipped maximum likelihood learning achieves the minimal KL divergence loss for infinite sample size.

Also note that in general, $D(P\|\tilde{P}_{BN})$ is *not* equal to $D(P\|P_{BN}) + D(P_{BN}\|\tilde{P}_{BN})$. In particular, the second term in Eqn. (46) is *not* equal to $D(P_{BN}\|\tilde{P}_{BN})$, since $P_{BN}(PaX_j = \mathbf{u})$ is (in general) not equal to $P(PaX_j = \mathbf{u})$. (In contrast, for log-linear models/undirected graphical models a decomposition of the KL-divergence does hold.[17])

E.3 Proofs of Lemmas 21, 22, 23, 24, 25

We will first state and prove two lemmas that are used to subsequently prove lemmas 21, 22, 23, 24, 25.

Lemma 27 *For any* $\theta^{(1)}_{1:v}, \theta^{(2)}_{1:v} \in [0,1]^v, \sum_{i=1}^v \theta^{(1)}_i = 1, \sum_{i=1}^v \theta^{(2)}_i = 1$*, we have*

$$D(\theta^{(1)}_{1:v}\|\theta^{(2)}_{1:v}) \leq \sum_{i=1}^{v} \frac{(\theta^{(1)}_i - \theta^{(2)}_i)^2}{\theta^{(2)}_i}.$$

Proof *We use the concavity of the* log *function, to upper bound it with a tangent line at* $\theta^{(2)}_i$*, which gives the following inequality:*

$$\log\theta^{(1)}_i \leq \log\theta^{(2)}_i + \frac{1}{\theta^{(2)}_i}(\theta^{(1)}_i - \theta^{(2)}_i). \tag{50}$$

Substituting Eqn. (50) into the definition of $D(\theta^{(1)}_{1:v}\|\theta^{(2)}_{1:v})$ *gives us:*

$$D(\theta^{(1)}_{1:v}\|\theta^{(2)}_{1:v}) \leq \sum_{i=1}^{v} \frac{\theta^{(1)}_i}{\theta^{(2)}_i}(\theta^{(1)}_i - \theta^{(2)}_i).$$

Adding $0 = \sum_{i=1}^v \frac{-\theta^{(2)}_i}{\theta^{(2)}_i}(\theta^{(1)}_i - \theta^{(2)}_i)$ *to the right hand side gives:*

$$D(\theta^{(1)}_{1:v}\|\theta^{(2)}_{1:v}) \leq \sum_{i=1}^{v} \frac{1}{\theta^{(2)}_i}(\theta^{(1)}_i - \theta^{(2)}_i)^2,$$

which proves the theorem. ■

Lemma 28 *For any v-valued multinomial distributions with means* $\theta^{(1)}_{1:v} \in [0,1]^v$ *and* $\theta^{(2)}_{1:v} \in [\gamma, 1-\gamma]^v$*, with* $\gamma < \frac{1}{v}$*, we have*

$$D(\theta^{(1)}_{1:v}\|\theta^{(2)}_{1:v}) \leq \sum_{i=1}^{v} \frac{(\theta^{(1)}_i - \theta^{(2)}_i)^2}{\gamma}.$$

Proof *Immediately from Lemma 27, since* $\frac{1}{\theta^{(2)}_i} \leq \frac{1}{\gamma}$. ■

17. Let P be any distribution, let $\{P_\theta\}$ be a family of log-linear models parameterized by θ, let $\theta^* = \arg\min_\theta D(P\|P_\theta)$. Then we do have that $D(P\|P_\theta) = D(P\|P_{\theta^*}) + D(P_{\theta^*}\|P_\theta)$. The proof relies on the fact that θ^* is such that $E_P[\eta_i] = E_{P_{\theta^*}}[\eta_i], \forall i$, with η_i the natural parameters of the log-linear model (see, for example, Kullback (1959)). Due to the local normalization constraints, this is not true in BN's.

Proof [Lemma 21] Let $\hat{\theta}_{1:v}$ be the unclipped sample means. The triangle inequality gives for any $i \in \{1, \cdots, v\}$:

$$|\theta_i^* - \tilde{\theta}_i| \leq |\theta_i^* - \hat{\theta}_i| + |\hat{\theta}_i - \tilde{\theta}_i|. \tag{51}$$

From the Hoeffding inequality and the Union bound we have that for all $i \in \{1, \cdots, v\}$ for

$$|\theta_i^* - \hat{\theta}_i| \leq \varepsilon' \tag{52}$$

to hold w.p. $1-\delta$, it suffices that

$$m \geq \frac{1}{2(\varepsilon')^2} \log \frac{2v}{\delta}. \tag{53}$$

Since $\tilde{\theta}_{1:v}$ are obtained by clipping $\hat{\theta}_{1:v}$ into $[\gamma, 1-\gamma]$ (for now γ is a free parameter, which will soon be matched to the clipping choice of $\frac{\varepsilon}{4v^3}$ of the lemma), we have that (see introduction of previous section)

$$|\hat{\theta}_i - \tilde{\theta}_i| \leq v\gamma. \tag{54}$$

Using Lemma 28 and then Eqn. (51), (52), and (54) we have that

$$D(\theta_{1:v} \| \tilde{\theta}_{1:v}) \leq \sum_{i=1}^{v} \frac{(\theta_i^* - \tilde{\theta}_i)^2}{\gamma} \leq v \frac{(\varepsilon' + v\gamma)^2}{\gamma} \tag{55}$$

holds w.p. $1-\delta$ if m satisfies Eqn. (53). The choice of $\gamma = \frac{\varepsilon'}{v}$ minimizes the right hand side of Eqn. (55), and gives us that

$$D(\theta_{1:v} \| \tilde{\theta}_{1:v}) \leq 4v^2 \varepsilon'. \tag{56}$$

Now choosing $\varepsilon' = \frac{\varepsilon}{4v^2}$ (corresponding to $\gamma = \frac{\varepsilon}{4v^3}$) gives us that for

$$D(\theta_{1:v} \| \tilde{\theta}_{1:v}) \leq \varepsilon$$

to hold w.p. $1-\delta$ it suffices that

$$m \geq \frac{8v^4}{\varepsilon^2} \log \frac{2v}{\delta},$$

which proves the lemma. ■

Proof [Lemma 22] We have

$$\begin{aligned}
D(\theta_{1:v}^{(1)} \| \theta_{1:v}^{(2)}) &= \sum_{i=1}^{v} \theta_i^{(1)} \log \frac{\theta_i^{(1)}}{\theta_i^{(2)}} \\
&\leq \max_i \log \frac{\theta_i^{(1)}}{\theta_i^{(2)}} \\
&\leq \max_i \log \frac{1}{\theta_i^{(2)}} \\
&\leq \max_{y \in [\gamma, 1-\gamma]} \log \frac{1}{y} \\
&= \log \frac{1}{\gamma},
\end{aligned}$$

which proves the lemma. ■

Proof [Lemma 23] Let $\hat{p} = \frac{m_H}{m}$, then

$$Pr(m_H \leq \frac{pm}{2}) = Pr(\hat{p} \leq \frac{p}{2}) = Pr(\frac{p-\hat{p}}{\sqrt{p}} \geq \frac{\sqrt{p}}{2}).$$

Applying the (multiplicative) Chernoff bound gives

$$Pr(m_H \leq \frac{pm}{2}) \leq \exp(\frac{-pm}{8}) = \delta,$$

where the last equality defines δ. Solving the last equation for m shows that $m = \frac{8}{p}\log\frac{1}{\delta}$ samples are sufficient to guarantee $Pr(m_H > \frac{pm}{2}) \geq 1-\delta$, which is the statement of the lemma. ■

Proof [Lemma 24] Below let $\theta_i = P(X_{k+1} = i|X_{1:k} = \mathbf{u})$, let $\tilde{\theta}_i = \tilde{P}(X_{k+1} = i|X_{1:k} = \mathbf{u})$, and let $\bar{v} = |\mathrm{val}(\mathrm{X_{k+1}})|$. We split the proof into 2 cases

1. $\frac{\varepsilon}{v^{k/2}\sqrt{P(X_{1:k}=\mathbf{u})}} \geq \log\frac{4v^3}{\varepsilon}$ This case is trivial, since by Lemma 22 we have that $D(\theta_{1:\bar{v}}\|\tilde{\theta}_{1:\bar{v}}) \leq \log\frac{4\bar{v}^3}{\varepsilon} \leq \log\frac{4v^3}{\varepsilon}$ and the statement of the lemma is trivially implied, for all $\tilde{\theta}_{1:\bar{v}} \in [\frac{4\bar{v}^3}{\varepsilon}, 1-\frac{4\bar{v}^3}{\varepsilon}]^{\bar{v}}$, so $m = 0$ samples is sufficient.

2. $\frac{\varepsilon}{v^{k/2}\sqrt{P(X_{1:k}=\mathbf{u})}} < \log\frac{4v^3}{\varepsilon}$ Let $m_{\mathbf{u}}$ be the number of samples for which $X_{1:k} = \mathbf{u}$. Then (using Lemma 21 and $\bar{v} \leq v$) a number of samples

$$m_{\mathbf{u}} \geq \frac{8v^4\ v^k P(X_{1:k}=\mathbf{u})}{\varepsilon^2}\log\frac{2v}{\delta'} \tag{57}$$

is sufficient to guarantee that $D(\theta_{1:\bar{v}}\|\tilde{\theta}_{1:\bar{v}}) \leq \frac{\varepsilon}{v^{k/2}\sqrt{P(X_{1:k}=\mathbf{u})}}$ with probability $1-\delta'$. To obtain, with probability $1-\delta''$, at least $m_{\mathbf{u}}$ samples from $P(X_{1:k+1})$ for which $X_{1:k} = \mathbf{u}$, it suffices that the total number of samples m from $P(X_{1:k+1})$ satisfies

$$m \geq \max\{\frac{8}{P(X_{1:k}=\mathbf{u})}\log\frac{1}{\delta''}, \frac{2m_{\mathbf{u}}}{P(X_{1:k}=\mathbf{u})}\},$$

where we used Lemma 23. Using $P(X_{1:k}=\mathbf{u}) \geq \frac{\varepsilon^2}{v^k\log^2\frac{4v^3}{\varepsilon}}$ (we are in case 2) and (57), and setting $\delta' = \delta/2$, $\delta'' = \delta/2$, gives the statement of the lemma.

■

Proof [Lemma 25] Using Lemma 24 and the union bound over all instantiations $\mathbf{u}$ of $X_{1:k}$ (there are at most v^k instantiations) we get that for

$$\forall \mathbf{u} \in \mathrm{val}(\mathrm{X_{1:k}})\ \ D(P(X_{k+1}|X_{1:k}=\mathbf{u})\|\tilde{P}(X_{k+1}|X_{1:k}=\mathbf{u})) \leq \frac{\varepsilon}{v^{k/2}\sqrt{P(X_{1:k}=\mathbf{u})}} \tag{58}$$

to hold with probability $1-\delta$, it suffices that

$$m \geq \frac{16v^{4+k}\log^2\frac{4v^3}{\varepsilon}}{\varepsilon^2}\log\frac{4v^{k+1}}{\delta}. \tag{59}$$

So we have that the following inequalities hold w.p. $1-\delta$ if m satisfies Eqn. (59):

$$\begin{aligned}
&\sum_{\mathbf{u}\in\mathrm{val}(\mathrm{X}_{1:k})} P(X_{1:k}=\mathbf{u})D(P(X_{k+1}|X_{1:k}=\mathbf{u})\|\tilde{P}(X_{k+1}|X_{1:k}=\mathbf{u}))\\
&\leq \sum_{\mathbf{u}} P(X_{1:k}=\mathbf{u})\frac{\varepsilon}{v^{\frac{k}{2}}\sqrt{P(X_{1:k}=\mathbf{u})}}\\
&\leq \sum_{\mathbf{u}} \varepsilon\frac{\sqrt{P(X_{1:k}=\mathbf{u})}}{v^{k/2}}\\
&\leq \varepsilon,
\end{aligned}$$

where we used in order: Eqn. (58), simplification, Cauchy-Schwarz. The last inequality together with the condition in Eqn. (59) prove the lemma. ■

Appendix F. Proofs for Section 4

In this section we give formal proofs of all theorems, propositions and lemmas appearing in Section 4.

F.1 Proof of Lemma 12

We first prove the following lemma.

Lemma 29 *Let any $\varepsilon > 0, \delta > 0$ be given. Let any $\lambda \in (0,1)$ be given. Let $\{X_i\}_{i=1}^m$ be i.i.d. Bernoulli(ϕ) random variables, where $\lambda \leq \phi \leq 1-\lambda$. Let $\hat{\phi} = \frac{1}{m}\sum_{i=1}^m X_i$. Then for*

$$|\phi\log\phi - \hat{\phi}\log\hat{\phi}| \leq \varepsilon$$

to hold w.p. $1-\delta$, it suffices that

$$m \geq \max\{\frac{2}{\lambda^2}\log\frac{2}{\delta}, \frac{2}{\lambda^2\varepsilon^2}\log\frac{2}{\delta}\}.$$

Proof *From the Hoeffding inequality we have that for*

$$|\phi - \hat{\phi}| \leq \varepsilon'$$

to hold w.p. $1-\delta$ it suffices that

$$m \geq \frac{1}{2\varepsilon'^2}\log\frac{2}{\delta}. \tag{60}$$

Now since the function $f(x) = x\log x$ is Lipschitz with Lipschitz-constant smaller than $\max\{1,|\log(\lambda-\varepsilon')|\}$ *over the interval $[\lambda-\varepsilon',1]$, we have that for*

$$|\phi\log\phi - \hat{\phi}\log\hat{\phi}| \leq \varepsilon'\max\{1,|\log(\lambda-\varepsilon')|\}$$

to hold w.p. $1-\delta$, it suffices that m satisfies Eqn. (60). If we choose ε' such that $\varepsilon' \leq \lambda/2$ we get

$$|\phi \log \phi - \hat{\phi} \log \hat{\phi}| \leq \varepsilon' \max\{1, |\log(\lambda/2)|\}.$$

To ensure the right hand side is smaller than ε, it suffices that the following three conditions are satisfied:

$$\begin{aligned} \varepsilon' &\leq \frac{\lambda}{2}, \\ \varepsilon' &\leq \varepsilon, \\ \varepsilon' &\leq \varepsilon\lambda/2 \leq \varepsilon/|\log \frac{\lambda}{2}|. \end{aligned}$$

The last inequality holds since $\lambda \in (0,1)$. Since $\lambda \in (0,1)$ we can simplify this to the following two conditions:

$$\begin{aligned} \varepsilon' &\leq \frac{\lambda}{2}, \\ \varepsilon' &\leq \varepsilon\lambda/2. \end{aligned}$$

Substituting this into Eqn. (60) gives us the condition for m as in the statement of the lemma. ∎

Proof [Lemma 12] We abbreviate $P(\mathbf{X}=\mathbf{x}, \mathbf{Y}=\mathbf{y})$ as $P(\mathbf{x},\mathbf{y})$ and similarly for $\hat{P}, \mathbf{x}, \mathbf{y}, \mathbf{x}|\mathbf{y}$. We abbreviate $\sum_{\mathbf{x}\in \mathrm{val}(\mathbf{X})}$ by $\sum_{\mathbf{x}}$ and similarly for $\mathbf{y}$.

$$\begin{aligned} \left|H(\mathbf{X}|\mathbf{Y}) - \widehat{H}(\mathbf{X}|\mathbf{Y})\right| &= \left|\sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x},\mathbf{y}) \log P(\mathbf{x}|\mathbf{y}) - \sum_{\mathbf{x},\mathbf{y}} \hat{P}(\mathbf{x},\mathbf{y}) \log \hat{P}(\mathbf{x}|\mathbf{y})\right| \\ &= \left|\sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x},\mathbf{y}) \log P(\mathbf{x},\mathbf{y}) - \sum_{\mathbf{y}} P(\mathbf{y}) \log P(\mathbf{y})\right. \\ &\quad \left. - \sum_{\mathbf{x},\mathbf{y}} \hat{P}(\mathbf{x},\mathbf{y}) \log \hat{P}(\mathbf{x},\mathbf{y}) + \sum_{\mathbf{y}} \hat{P}(\mathbf{y}) \log \hat{P}(\mathbf{y})\right| \\ &\leq \sum_{\mathbf{x},\mathbf{y}} \left|P(\mathbf{x},\mathbf{y}) \log P(\mathbf{x},\mathbf{y}) - \hat{P}(\mathbf{x},\mathbf{y}) \log \hat{P}(\mathbf{x},\mathbf{y})\right| \\ &\quad + \sum_{\mathbf{y}} \left|P(\mathbf{y}) \log P(\mathbf{y}) - \hat{P}(\mathbf{y}) \log \hat{P}(\mathbf{y})\right| \end{aligned}$$

Now using Lemma 29 (and the Union bound) we get that for

$$\left|H(\mathbf{X}|\mathbf{Y}) - \widehat{H}(\mathbf{X}|\mathbf{Y})\right| \leq |\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|\varepsilon' + |\mathrm{val}(\mathbf{Y})|\varepsilon'$$

to hold w.p. $1 - |\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|\delta' - |\mathrm{val}(\mathbf{X})|\delta'$, it suffices that

$$m \geq \max\{\frac{2}{\lambda^2 \varepsilon'^2} \log \frac{2}{\delta'}, \frac{2}{\lambda^2} \log \frac{2}{\delta'}\}.$$

Choosing $\varepsilon = \varepsilon'/(2|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|)$ and $\delta = \delta'/(2|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|)$ gives that for

$$|H(\mathbf{X}|\mathbf{Y}) - \widehat{H}(\mathbf{X}|\mathbf{Y})| \leq \varepsilon$$

to hold with probability $1-\delta$, it suffices to have

$$m \geq \max\{\frac{8|\mathrm{val}(\mathbf{X})|^2|\mathrm{val}(\mathbf{Y})|^2}{\lambda^2\varepsilon^2}\log\frac{4|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|}{\delta}, \frac{2}{\lambda^2}\log\frac{4|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|}{\delta}\}. \tag{61}$$

Now since for any two distributions P and $\hat{P}$ we have $|H(\mathbf{X}|\mathbf{Y}) - \widehat{H}(\mathbf{X}|\mathbf{Y})| \leq \log|\mathrm{val}(\mathbf{X})| \leq 2|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|$, we have that for any $\varepsilon \geq 2|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|$ the statement of the lemma holds trivially independent of the number of samples m. Thus we can simplify the conditions on m in Eqn. (61) to one condition:

$$m \geq \frac{8|\mathrm{val}(\mathbf{X})|^2|\mathrm{val}(\mathbf{Y})|^2}{\lambda^2\varepsilon^2}\log\frac{4|\mathrm{val}(\mathbf{X})||\mathrm{val}(\mathbf{Y})|}{\delta},$$

which proves the lemma. ■

F.2 Proof of Lemma 13

We abbreviate $P(\mathbf{X}=\mathbf{x})$ as $P(\mathbf{x})$ and similarly for other variables.
Proof [Lemma 13] Using Eqn. (14) and the definition of conditional entropy we get that

$$\sum_{\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y}} P(\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y})\log P(\mathbf{x}|\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y}) - \sum_{\mathbf{x},\mathbf{u},\mathbf{w}} P(\mathbf{x},\mathbf{u},\mathbf{w})\log P(\mathbf{x}|\mathbf{u},\mathbf{w}) \leq \varepsilon.$$

We can rewrite this as

$$\sum_{\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y}} P(\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y})\log\frac{P(\mathbf{x}|\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y})}{P(\mathbf{x}|\mathbf{u},\mathbf{w})} \leq \varepsilon.$$

Now using Eqn. (13) ($\mathbf{U}\cup\mathbf{V}$ is the Markov blanket of $\mathbf{X}$) gives us

$$\sum_{\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y}} P(\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y})\log\frac{P(\mathbf{x}|\mathbf{u},\mathbf{v})}{P(\mathbf{x}|\mathbf{u},\mathbf{w})} \leq \varepsilon.$$

We can simplify this to

$$\sum_{\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w}} P(\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w})\log\frac{P(\mathbf{x}|\mathbf{u},\mathbf{v})}{P(\mathbf{x}|\mathbf{u},\mathbf{w})} \leq \varepsilon.$$

Using the definition of conditional probability and Eqn. (13) ($\mathbf{U}\cup\mathbf{V}$ is the Markov blanket of $\mathbf{X}$) we get

$$\sum_{\mathbf{u},\mathbf{v},\mathbf{w}} P(\mathbf{u},\mathbf{v},\mathbf{w})\sum_{\mathbf{x}} P(\mathbf{x}|\mathbf{u},\mathbf{v})\log\frac{P(\mathbf{x}|\mathbf{u},\mathbf{v})}{P(\mathbf{x}|\mathbf{u},\mathbf{w})} \leq \varepsilon.$$

Now since $\lambda_1 \leq P(\mathbf{u},\mathbf{v},\mathbf{w})$ and each term $\sum_{\mathbf{x}} P(\mathbf{x}|\mathbf{u},\mathbf{v})\log\frac{P(\mathbf{x}|\mathbf{u},\mathbf{v})}{P(\mathbf{x}|\mathbf{u},\mathbf{w})}$ is positive (it's a KL-divergence) we get that for all $\mathbf{u},\mathbf{v},\mathbf{w}$

$$\sum_{\mathbf{x}} P(\mathbf{x}|\mathbf{u},\mathbf{v})\log\frac{P(\mathbf{x}|\mathbf{u},\mathbf{v})}{P(\mathbf{x}|\mathbf{u},\mathbf{w})} \leq \frac{\varepsilon}{\lambda_1}.$$

The left hand side of this equation is the KL-divergence between a distribution $Q_{\mathbf{u},\mathbf{v},\mathbf{w}}(\mathbf{X}) = P(\mathbf{X}|\mathbf{U} = \mathbf{u}, \mathbf{V} = \mathbf{v})$ and a distribution $\hat{Q}_{\mathbf{u},\mathbf{v},\mathbf{w}}(\mathbf{X}) = P(\mathbf{X}|\mathbf{U} = \mathbf{u}, \mathbf{W} = \mathbf{w})$. Now using the KL-divergence property that $\frac{1}{2}(\sum_{\mathbf{x}} |P_1(\mathbf{x}) - P_2(\mathbf{x})|)^2 \leq D(P_1 \| P_2)$ (see, for example, Cover and Thomas, 1991, p. 300), we get that for all $\mathbf{u},\mathbf{v},\mathbf{w}$

$$\frac{1}{2}(\sum_{\mathbf{x}} |P(\mathbf{x}|\mathbf{u},\mathbf{v}) - P(\mathbf{x}|\mathbf{u},\mathbf{w})|)^2 \leq \frac{\varepsilon}{\lambda_1}.$$

As a consequence, we have for all $\mathbf{x},\mathbf{u},\mathbf{v},\mathbf{w}$ that

$$|P(\mathbf{x}|\mathbf{u},\mathbf{v}) - P(\mathbf{x}|\mathbf{u},\mathbf{w})| \leq \sqrt{2\frac{\varepsilon}{\lambda_1}}.$$

Now since $\lambda_2 \leq P(\mathbf{x}|\mathbf{u},\mathbf{v})$ and $\lambda_2 \leq P(\mathbf{x}|\mathbf{u},\mathbf{w})$ we have that

$$|\log P(\mathbf{x}|\mathbf{u},\mathbf{v}) - \log P(\mathbf{x}|\mathbf{u},\mathbf{w})| \leq \frac{\sqrt{2\varepsilon}}{\lambda_2\sqrt{\lambda_1}}.$$

Now using Eqn. (13) ($\mathbf{U} \cup \mathbf{V}$ is the Markov blanket of $\mathbf{X}$) to substitute $P(\mathbf{x}|\mathbf{u},\mathbf{v})$ by $P(\mathbf{x}|\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{y})$, we obtain Eqn. (15). ∎

F.3 Proof of Theorem 14

Proof [Theorem 14] There are $O(kn^k bn^b)$ ⟨candidate factor, candidate Markov blanket⟩ pairs, each with $O(v^{k+b})$ different instantiations. Collecting the required empirical probabilities from the data takes $O(kn^k bn^b v^{k+b} + mkn^k bn^b(k+b))$. (Similar reasoning as in the proof of Theorem 5.) Computing the empirical entropies from the empirical probabilities takes $O(kn^k bn^b v^{k+b})$. There are $O(kn^k)$ actual factors computed. From (the proof of) Theorem 5, we have that this takes $O(kn^k(m(k+b) + 2^k v^k))$. Putting it all together gives us an upper bound on the running time of

$$\begin{aligned} &O\big(kn^k bn^b v^{k+b} + mkn^k bn^b (k+b) + kn^k bn^b v^{k+b} \\ &+ \quad kn^k(m(k+b) + 2^k v^k)\big). \end{aligned}$$

After simplification we get a running time of

$$O\big(kn^k bn^b v^{k+b} + mkn^k bn^b (k+b) + kn^k 2^k v^k\big).$$

∎

F.4 Proof of Theorem 15

Proof [Theorem 15] Let $\mathcal{C}$, $\mathcal{Y}$ be defined as in Eqn. (16) and Eqn. (17) of the structure learning algorithm description. For all $\mathbf{C}_j^* \in \mathcal{C}$ we have by assumption $|\mathrm{val}(\mathbf{C}_j^*)| \leq v^k$. For all $\mathbf{Y} \in \mathcal{Y}$ we have $|\mathrm{val}(\mathbf{Y})| \leq v^b$. Also note that $P(\mathbf{C}_j^* = \mathbf{c}_j^*, \mathbf{Y} = \mathbf{y}) \geq \frac{1}{\gamma^{k+b}}$. Using Lemma 12 we get that for any $\mathbf{C}_j^* \in \mathcal{C}, \mathbf{Y} \in \mathcal{Y}$ for

$$\left|H(\mathbf{C}_j^*|\mathbf{Y}) - \widehat{H}(\mathbf{C}_j^*|\mathbf{Y})\right| \leq \varepsilon' \tag{62}$$

to hold with probability $1-\delta'$ it suffices that

$$m \geq 8\frac{v^{2k}v^{2b}}{\gamma^{2b+2k}\varepsilon'^2}\log 4\frac{v^k v^b}{\delta'}. \tag{63}$$

Taking the union bound we get that for Eqn. (62) to hold for all $\mathbf{C}_j^* \in \mathcal{C}$ and for all $\mathbf{Y} \in \mathcal{Y}$ with probability $1-|\mathcal{C}||\mathcal{Y}|\delta'$ it suffices that m satisfies Eqn. (63).

For $\widehat{\mathrm{MB}}(\mathbf{C}_j^*) = \arg\min_{\mathbf{Y}\in\mathcal{Y},\mathbf{Y}\cap\mathbf{C}_j^*=\emptyset} \widehat{H}(\mathbf{C}_j^*|\mathbf{Y})$ we have $\widehat{\mathrm{H}}(\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)) \leq \widehat{\mathrm{H}}(\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*))$. Combining this with Eqn. (62) gives us

$$H(\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)) \leq H(\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)) + 2\varepsilon'. \tag{64}$$

From Lemma 13 we have that Eqn. (64) implies that

$$\begin{aligned} |\log P(\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)) - \log P(\mathbf{C}_j^*|\mathcal{X}-\mathbf{C}_j^*)| &\leq \frac{\sqrt{4\varepsilon'}}{\gamma^k\sqrt{\gamma^{2b}}} \\ &= \frac{2\sqrt{\varepsilon'}}{\gamma^{k+b}}. \end{aligned} \tag{65}$$

Now from Lemma 18 we have that for

$$|\log P(\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)) - \log \hat{P}(\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*))| \leq \frac{2\sqrt{\varepsilon'}}{\gamma^{k+b}} \tag{66}$$

to hold for all instantiations $\mathbf{c}_j^* \in \mathrm{val}(\mathbf{C}_j^*)$ with probability $1-\delta''$, it suffices that

$$m \geq \frac{(1+\frac{\sqrt{\varepsilon'}}{\gamma^{k+b}})^2}{2\gamma^{2k+2b}(\frac{\sqrt{\varepsilon'}}{\gamma^{k+b}})^2}\log\frac{4v^{k+b}}{\delta''}. \tag{67}$$

Using the Union bound, we get that for Eqn. (66) to hold for all $\mathbf{C}_j^* \in \mathcal{C}$ with probability $1-|\mathcal{C}|\delta''$, it suffices that m satisfies Eqn. (67). Or after simplification (and slightly loosening using $\gamma < 1$), we get the condition

$$m \geq \frac{(1+2\sqrt{\varepsilon'})^2}{2\gamma^{2k+2b}\varepsilon'}\log\frac{4v^{k+b}}{\delta''}. \tag{68}$$

Combining Eqn. (66) and Eqn. (65) gives us

$$|\log P(\mathbf{C}_j^*|\mathcal{X}-\mathbf{C}_j^*) - \log \hat{P}(\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*))| \leq \frac{4\sqrt{\varepsilon'}}{\gamma^{k+b}}.$$

From Lemma 19 we have that this implies

$$|\log f^*_{\mathbf{C}_j^*|\mathcal{X}-\mathbf{C}_j^*}(\mathbf{c}_j^*) - \log \hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)| \leq 2^{k+2}\frac{\sqrt{\varepsilon'}}{\gamma^{k+b}}.$$

Now choosing $\varepsilon' = (\frac{\varepsilon}{2^{k+2}})^2\frac{\gamma^{2k+2b}}{2^{2k+4}}$ gives us that

$$|\log f^*_{\mathbf{C}_j^*|\mathcal{X}-\mathbf{C}_j^*}(\mathbf{c}_j^*) - \log \hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)| \leq \frac{\varepsilon}{2^{k+2}}.$$

The clipping to one of factor entries $\hat{f}^*_{\mathbf{C}^*_j|\widehat{\mathrm{MB}}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)$ satisfying $|\log \hat{f}^*_{\mathbf{C}^*_j|\widehat{\mathrm{MB}}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)| \leq \frac{\varepsilon}{2^{k+2}}$ introduces at most an additional error of $\frac{\varepsilon}{2^{k+2}}$. Thus after the clipping we have,

$$|\log f^*_{\mathbf{C}^*_j|\mathcal{X}-\mathbf{C}^*_j}(\mathbf{c}^*_j) - \log \hat{f}^*_{\mathbf{C}^*_j|\widehat{\mathrm{MB}}(\mathbf{C}^*_j)}(\mathbf{c}^*_j)| \leq \frac{\varepsilon}{2^{k+1}}, \tag{69}$$

for all canonical factors of P. We also have that all candidate factors that are not present in the canonical form of the true distribution P will now have been removed and do not contribute to $\tilde{P}$. (By our assumption on b the algorithm considered large enough Markov blanket candidates to include the true Markov blanket. Such a large enough b for these factors (which can be larger than the maximum Markov blanket size for factors present in the distribution) is important. Trivial (all-ones) canonical factors computed using their Markov blanket require a true Markov blanket to be all-ones.)

So far we have shown that Eqn. (69) holds with probability $1-|\mathcal{C}||\mathcal{Y}|\delta'-|\mathcal{C}|\delta''$ if m satisfies both Eqn. (63) and Eqn. (68). Or, after substituting in the choice of ε', if the following hold

$$\begin{aligned} m &\geq \frac{2^{8k+19}v^{2k+2b}}{\gamma^{6k+6b}\varepsilon^4}\log\frac{4v^{k+b}}{\delta'}, \\ m &\geq 2^{4k+7}\frac{(1+2\frac{\varepsilon\gamma^{k+b}}{2^{2k+4}})^2}{\gamma^{4k+4b}\varepsilon^2}\log\frac{4v^{k+b}}{\delta''}. \end{aligned}$$

So choosing $\delta'=\delta''=\delta/(2|\mathcal{C}||\mathcal{Y}|)$, we have that for Eqn. (69) to hold with probability $1-\delta$, it suffices that

$$m \geq (1+\frac{\varepsilon\gamma^{k+b}}{2^{2k+3}})^2\frac{v^{2k+2b}2^{8k+19}}{\gamma^{6k+6b}\min\{\varepsilon^2,\varepsilon^4\}}\log\frac{8|\mathcal{C}||\mathcal{Y}|v^{k+b}}{\delta}.$$

Now using the fact that $|\mathcal{C}|\leq kn^k$ and $|\mathcal{Y}|\leq bn^b$ we obtain the following result: with probability $1-\delta$, Eqn. (69) holds for all non-trivial canonical factors in the target distribution if m satisfies the condition on m in the theorem, namely Eqn. (19). Moreover (recall the clipping procedure removed all candidate factors with scope less than k and Markov blanket size less than b that are not present in the canonical form of the true distribution P) we have that zero error is incurred on all other factors. Thus (after using Lemma 20) we have that

$$D(P\|\hat{P})+D(\hat{P}\|P)\leq 2J^*\frac{\varepsilon}{2^{k+1}}\leq J\varepsilon.$$

The second inequality follows since $J^*\leq 2^kJ$.

■

F.5 Proof of Theorem 16

Proof [Theorem 16] From Proposition 4 we have that

$$Q(\mathbf{x})=\frac{1}{Z}\prod_j f^*_{\mathbf{D}^*_j|\mathrm{MB}(\mathbf{D}^*_j)}(\mathbf{d}^*_j).$$

We can rewrite this product as follows:

$$Q(\mathbf{x}) = \frac{1}{Z} \prod_{j:|\mathbf{D}_j^*|\le k, |\mathrm{MB}(\mathbf{D}_j^*)|\le b} f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*) \prod_{j:|\mathbf{D}_j^*|\le k, |\mathrm{MB}(\mathbf{D}_j^*)|> b} f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*) \prod_{j:|\mathbf{D}_j^*|> k} f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*) \tag{70}$$

The learned distribution $\tilde{P} = \frac{1}{\tilde{Z}} \prod_{j=1}^{J^*} \hat{f}^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)$, can be rewritten as

$$\tilde{P}(\mathbf{x}) = \frac{1}{\tilde{Z}} \prod_{j:|\mathbf{D}_j^*|\le k, |\mathrm{MB}(\mathbf{D}_j^*)|\le b} \hat{f}^*{}_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*) \prod_{j:|\mathbf{D}_j^*|\le k, |\mathrm{MB}(\mathbf{D}_j^*)|> b} \hat{f}^*{}_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*) \prod_{j:\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k, |\mathrm{MB}(\mathbf{C}_j^*)|\le b} \hat{f}^*{}_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) \prod_{j:\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k, |\mathrm{MB}(\mathbf{C}_j^*)|> b} \hat{f}^*{}_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*). \tag{71}$$

Using Eqn. (70), Eqn. (71), the fact that for all $\mathbf{C}_j^* \notin \{\mathbf{D}_k^*\}_k$ we have that the canonical factor is trivial, namely $\log f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*) = 0$ (and adding and subtracting the same terms) we get: $D(Q\|\tilde{P}) + D(\tilde{P}\|Q) =$

$$E_{\mathbf{X}\sim Q}\Big(\sum_{j:\substack{|\mathbf{D}_j^*|\le k \\ |\mathrm{MB}(\mathbf{D}_j^*)|\le b}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*{}_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) - E_{\mathbf{X}\sim\tilde{P}}\Big(\sum_{j:\substack{|\mathbf{D}_j^*|\le k \\ |\mathrm{MB}(\mathbf{D}_j^*)|\le b}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*{}_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) \tag{72}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{j:\substack{|\mathbf{D}_j^*|\le k \\ |\mathrm{MB}(\mathbf{D}_j^*)|> b}} \log \frac{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*{}_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) - E_{\mathbf{X}\sim\tilde{P}}\Big(\sum_{j:\substack{|\mathbf{D}_j^*|\le k \\ |\mathrm{MB}(\mathbf{D}_j^*)|> b}} \log \frac{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{\hat{f}^*{}_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) \tag{73}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{j:\substack{|\mathbf{D}_j^*|\le k \\ |\mathrm{MB}(\mathbf{D}_j^*)|> b}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) - E_{\mathbf{X}\sim\tilde{P}}\Big(\sum_{j:\substack{|\mathbf{D}_j^*|\le k \\ |\mathrm{MB}(\mathbf{D}_j^*)|> b}} \log \frac{f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}{f^*_{\mathbf{D}_j^*|\widehat{\mathrm{MB}}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)}\Big) \tag{74}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{j:|\mathbf{D}_j^*|>k} \log f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)\Big) - E_{\mathbf{X}\sim\tilde{P}}\Big(\sum_{j:|\mathbf{D}_j^*|>k} \log f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j^*)\Big) \tag{75}$$

$$-E_{\mathbf{X}\sim Q}\Big(\sum_{j:\substack{\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k \\ |\mathrm{MB}(\mathbf{C}_j^*)|\le b}} \log \hat{f}^*{}_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)\Big) + E_{\mathbf{X}\sim P}\Big(\sum_{j:\substack{\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k \\ |\mathrm{MB}(\mathbf{C}_j^*)|\le b}} \log \hat{f}^*{}_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)\Big) \tag{76}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{j:\substack{\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k \\ |\mathrm{MB}(\mathbf{C}_j^*)|> b}} \log \frac{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{\hat{f}^*{}_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) - E_{\mathbf{X}\sim P}\Big(\sum_{j:\substack{\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k \\ |\mathrm{MB}(\mathbf{C}_j^*)|> b}} \log \frac{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{\hat{f}^*{}_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) \tag{77}$$

$$+E_{\mathbf{X}\sim Q}\Big(\sum_{j:\substack{\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k \\ |\mathrm{MB}(\mathbf{C}_j^*)|> b}} \log \frac{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) - E_{\mathbf{X}\sim P}\Big(\sum_{j:\substack{\mathbf{C}_j^*\notin\{\mathbf{D}_k^*\}_k \\ |\mathrm{MB}(\mathbf{C}_j^*)|> b}} \log \frac{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{\mathrm{MB}}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}\Big) \tag{78}$$

$$+\log\frac{Z}{\tilde{Z}} + \log\frac{\tilde{Z}}{Z}. \tag{79}$$

Using the same reasoning as in the proof of Theorem 15 we obtain that for the sum of the terms in lines (72), (73), (76) and (77) to be bounded by $(J+|S|)\varepsilon$ with probability at least $1-\delta$, it suffices that m satisfies Eqn. (19). The additional term in the bound, namely $|S|\varepsilon$, is necessary to bound the error contribution of the terms in line (77).

The sum of the terms in lines (74) and (78) can be bounded by

$$2\sum_{\mathbf{C}_j^*\in\mathcal{C}\,:\,|\mathrm{MB}(\mathbf{C}_j^*)|>b} \max_{\mathbf{c}_j^*}\left|\log\frac{f^*_{\mathbf{C}_j^*|\mathrm{MB}(\mathbf{C}_j^*)}(\mathbf{c}_j^*)}{f^*_{\mathbf{C}_j^*|\widehat{MB}(\mathbf{C}_j^*)}(\mathbf{C}_j^*)}\right|$$

The sum of the terms in line (75) can be bounded by (recall $\mathrm{MB}(\cdot)$ is the true Markov blanket for the true distribution Q, thus $f^*_{\mathbf{D}_j^*}(\mathbf{d}_j)=f^*_{\mathbf{D}_j^*|\mathrm{MB}(\mathbf{D}_j^*)}(\mathbf{d}_j)$)

$$2\sum_{j:|\mathbf{D}_j^*|>k}\max_{\mathbf{d}_j^*}\left|\log f^*_{\mathbf{D}_j^*}(\mathbf{d}_j)\right|.$$

The two terms in line (79) sum to zero.

This establishes the theorem. ■

References

P. Abbeel, D. Koller, and A. Y. Ng. Learning factor graphs in polynomial time & sample complexity. In *Proc. UAI*, 2005.

N. Abe, J. Takeuchi, and M. Warmuth. Polynomial learnability of probabilistic concepts with respect to the Kullback-Leibler divergence. In *Proc. COLT*, 1991.

N. Abe, J. Takeuchi, and M. Warmuth. On the computational complexity of approximating probability distributions by probabilistic automata. *Machine Learning*, 1992.

F. Bach and M. Jordan. Thin junction trees. In *NIPS 14*, 2002.

F. Barahona. On the computational complexity of Ising spin glass models. *J. Phys. A*, 1982.

J. Besag. Efficiency of pseudo-likelihood estimation for simple Gaussian fields. *Biometrika*, 1974a.

J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*, 1974b.

J. Cheng, R. Greiner, J. Kelly, D. Bell, and W. Liu. Learning Bayesian networks from data: An information-theory based approach. *Artificial Intelligence Journal*, 2002.

D. M. Chickering. Learning Bayesian networks is NP-Complete. In D. Fisher and H.J. Lenz, editors, *Learning from Data: Artificial Intelligence and Statistics V*, pages 121–130. Springer-Verlag, 1996.

D. M. Chickering and C. Meek. Finding optimal Bayesian networks. In *Proc. UAI*, 2002.

D. M. Chickering, C. Meek, and D. Heckerman. Large-sample learning of Bayesian networks is hard. In *Proc. UAI*, 2003.

C. K. Chow and C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 1968.

F. Comets. On consistency of a class of estimators for exponential families of Markov random fields on the lattice. *Annals of Statistics*, 1992.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991.

R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer, 1999.

S. Dasgupta. The sample complexity of learning fixed structure Bayesian networks. *Machine Learning*, 1997.

S. Dasgupta. Learning polytrees. In *Proc. UAI*, 1999.

S. Della Pietra, V. J. Della Pietra, and J. D. Lafferty. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393, 1997.

A. Deshpande, M. N. Garofalakis, and M. I. Jordan. Efficient stepwise selection in decomposable models. In *Proc. UAI*, pages 128–135. Morgan Kaufmann Publishers Inc., 2001. ISBN 1-55860-800-1.

N. Friedman and Z. Yakhini. On the sample complexity of learning Bayesian networks. In *Proc. UAI*, 1996.

S. Geman and C. Graffigne. Markov random field image models and their applications to computer vision. In *Proc. of the International Congress of Mathematicians*, 1986.

C. J. Geyer and E. A. Thompson. Constrained Monte Carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society, Series B*, 1992.

B. Gidas. Consistency of maximum likelihood and pseudo-likelihood estimators for Gibbsian distributions. In W. Fleming and P.-L. Lions, editors, *Stochastic differential systems, stochastic control theory and applications*. Springer, New York, 1988.

X. Guyon and H. R. Künsch. Asymptotic comparison of estimators in the Ising model. In *Stochastic Models, Statistical Methods, and Algorithms in Image Analysis, Lecture Notes in Statistics*. Springer, Berlin, 1992.

J. M. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. Unpublished, 1971.

K. L. Höffgen. Learning and robust learning of product distributions. In *Proc. COLT*, 1993.

F. Huang and Y. Ogata. Generalized pseudo-likelihood estimates for Markov random fields on lattice. *Annals of the Institute of Statistical Mathematics*, 2002.

M. Jerrum and A. Sinclair. Polynomial-time approximation algorithms for the Ising model. *SIAM J. Comput.*, 1993.

C. Ji and L. Seymour. A consistent model selection procedure for Markov random fields based on penalized pseudolikelihood. *Annals of Applied Probability*, 1996.

D. Karger and N. Srebro. Learning Markov networks: maximum bounded tree-width graphs. In *Symposium on Discrete Algorithms*, pages 392–401, 2001.

F. R. Kschischang, B. J. Frey, and H. A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 2001.

S. Kullback. *Probability theory and statistics*. Wiley, 1959.

S. L. Lauritzen. *Graphical Models*. Oxford University Press, 1996.

F. M. Malvestuto. Approximating discrete probability distributions with decomposable models. *IEEE Transactions on Systems, Man and Cybernetics*, 1991.

A. McCallum. Efficiently inducing features of conditional random fields. In *Proc. UAI*, 2003.

C. Meek. Finding a path is harder than finding a tree. *Journal of Artificial Intelligence Research*, 15:383–389, 2001.

M. Narasimhan and J. Bilmes. PAC-learning bounded tree-width graphical models. In *Proc. UAI*, 2004.

A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. In *NIPS 14*, 2002.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search (second edition)*. MIT Press, 2000.

N. Srebro. Maximum likelihood bounded tree-width Markov networks. In *Proc. UAI*, 2001.

V. N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.

J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. Technical report, Mitsubishi Electric Reseach Laboratories, 2001.