Greedy Learning of Markov Network Structure

(Invited Paper)

Praneeth Netrapalli, Siddhartha Banerjee, Sujay Sanghavi, Sanjay Shakkottai
The University of Texas at Austin
Austin, Texas 78712
Email: {praneethn,sbanerjee,sanghavi,shakkott}@mail.utexas.edu

***Abstract*—Markov Random Fields (MRFs), a.k.a. Graphical Models, serve as popular models for networks in the social and biological sciences, as well as communications and signal processing. A central problem is one of structure learning or model selection: given samples from the MRF, determine the graph structure of the underlying distribution. When the MRF is not Gaussian (e.g. the Ising model) and contains cycles, structure learning is known to be NP hard even with infinite samples. Existing approaches typically focus either on specific parametric classes of models, or on the sub-class of graphs with bounded degree; the complexity of many of these methods grows quickly in the degree bound. We develop a simple new 'greedy' algorithm for learning the structure of graphical models of discrete random variables. It learns the Markov neighborhood of a node by sequentially adding to it the node that produces the highest reduction in conditional entropy. We provide a general sufficient condition for exact structure recovery (under conditions on the degree/girth/correlation decay), and study its sample and computational complexity. We then consider its implications for the Ising model, for which we establish a self-contained condition for exact structure recovery.**

I. Introduction

Markov Random Fields (MRF) are undirected graphical models which are used to encode conditional independence relations between random variables. At a more abstract level, a graphical model captures the dependencies between a collection of entities. Thus the nodes of a graphical model may represent people, genes, languages, processes, etc., while the graphical model illustrates certain conditional dependencies among them (for example, influence in a social network, physiological functionality in genetic networks, etc.). Often the knowledge of the underlying graph is not available beforehand, but must be inferred from certain observations of the system. In mathematical terms, these observations correspond to samples drawn from the underlying distribution. Thus, the core task of structure learning is that of inferring conditional dependencies between random variables from i.i.d samples drawn from their joint distribution. The importance of the MRF in understanding the underlying system makes structure learning an important primitive for studying such systems.

More specifically, an MRF is an undirected graph $G(V, E)$, where the vertex set $V = \{v_1, v_2, \ldots, v_p\}$ corresponds to a p-dimensional random variable $X = \{X_1, X_2, \ldots, X_p\}$ (whereby each vertex i is associated with variable X_i), and the edges encode the conditional dependencies between the random variables (this is explained in detail in Section II). A structure learning algorithm takes as input, samples drawn from the distribution of X, and outputs an estimate $\hat{G}$ of the underlying MRF. There are three primary yardsticks for a structure learning algorithm:- correctness, sample complexity and computational complexity. The three are interdependent, and in a sense an ideal structure learning algorithm is one which can learn any underlying graph on the nodes with high probability (or with probability of error less than some given δ, analogous to the PAC model of learning) with associated sample complexity and computational complexity polynomial in p and $\frac{1}{\delta}$. However, it is known that the general structure learning problem is a difficult problem, both in terms of sample complexity [1], [2] and computational complexity [3], [4]. Inspite of this, the practical importance of the problem has motivated a lot of work in this topic, and there are several approaches in the literature that, although not optimal, perform well (both in practice, and also theoretically) under some stronger constraints on the problem.

There are two fundamental ways to perform structure learning, corresponding to two different interpretations of a graphical model. Under certain conditions (given by the Hammersley-Clifford theorem [5]), the conditional independence view of a graphical model leads to a factorization of the joint probability mass function (or density) according to the cliques of the graph. *Parameter estimation techniques* [6] [7] utilize such a factorization of the distribution to learn the underlying graph. These techniques assume a certain form of the potential function, and thereby relate the structure learning problem to one of finding a sparse maximum likelihood estimator of a distribution from its samples. On the other hand, algorithms based on learning conditional independence relations between the variables, which we refer to as *comparison tests*, are potential agnostic, i.e., they do not need knowledge of the underlying parametrization to learn the graph. These methods are based on comparing all possible neighborhoods of a node to find one which has the 'maximum influence' on the node. In both cases, in order to learn the underlying graph accurately and efficiently, the algorithms need some assumptions on the underlying distribution and graph structure. There are several existing comparison test based methods [8] [9] [10], each with associated conditions under which they can learn the graph correctly.

In addition to the difference in underlying assumptions, there is another fundamental difference in the philosophy of the two approaches. The parameter estimation techniques tend to be 'bottom-up' approaches, whereby the algorithm is proposed first, based on some intuition regarding the system, and then subsequently it is analyzed and conditions are found for correctness and efficiency. On the other hand, the comparison-test techniques in literature tend to be designed with the aim of achieving some correctness requirements. As a result, comparison-test algorithms usually involve a computationally expensive search over all potential neighborhoods of a node, and this increases their computational complexity. In addition, although these algorithms make no assumptions on the parametrization of the distribution, they need to assume some properties of the graph in order to succeed (for example, the algorithm of Bresler et. al. [8] needs to know the maximum degree of the graph in order to learn it). Our contribution in this work is to propose a simple 'greedy', comparison-test based algorithm for learning MRF structure. As in any sub-optimal greedy algorithm, we can not always guarantee correctness, but are guaranteed low computational complexity. However, we are able to provide general sufficient conditions for the success of the algorithm for any graphical model, and show that these conditions are in fact satisfied by one specific graphical model of significance in literature: the pairwise symmetric binary model, or the Ising model.

Greedy comparison-tests for exact structure learning are however not completely new, and in fact one of the early successes in the field was in the form of a greedy algorithm. In their seminal paper, Chow and Liu [10] showed that if the MRF was a tree, then it could be learnt by a simple maximum spanning tree algorithm. However their method is crucially dependent on the underlying graph being a spanning tree (although recent results [11] have shown how it can be modified to learn general acyclic graphs), and fails as soon as the graph has loops. Our algorithm, in some sense, generalizes the Chow and Liu algorithm to a richer class of graphs. This is in spirit similar to the manner in which loopy belief propagation extends the dynamic programming paradigm from trees to loopy graphs. One notes however that unlike the Chow and Liu algorithm which searches for a globally optimal graph, ours is a locally greedy algorithm, whereby we learn the neighborhood of each node separately in a greedy manner.

The remaining paper are organized as follows. In Section II, we review graphical models and some results from information theory, and set up the structure learning problem. Our new structure learning algorithm, GreedyAlgorithm(ϵ), is given in Section III. Next, in Section IV, we develop a sufficient condition for the correctness of the algorithm for general graphs. To demonstrate the applicability of this condition, we translate it into equivalent conditions for learning an Ising model in Section V. We discuss future work and conclude in Section VI.

II. Preliminaries

In this section, we formally define a graphical model and set up the structure learning algorithm. In addition, as a foreshadow to our structure learning algorithm, we define conditional entropy, and state some of its properties which we use later. We also define a notion of 'empirical' conditional entropy which we later use as our test function, and state an important lemma from information theory that helps relate empirical entropy and empirical measures. For more details regarding graphical models, refer to [5], and for the information theoretic definitions, refer to [12].

First we establish some notation that we use throughout. We assume in this paper that the random vector X whose graph we are trying to learn is discrete valued. More specifically, we assume that X is an n-dimensional random vector $\{X_1, X_2, \ldots, X_n\}$, where each component X_i of X takes values in a finite set $\mathcal{X}$. We use the shorthand notation $P(x_i)$ to stand for $\mathbb{P}(X_i = x_i), x_i \in \mathcal{X}$, and similarly for a set $A \subseteq \{1, 2, \ldots, n\}$, we define $P(x_A) \triangleq \mathbb{P}(X_A = x_A), x_a \in \mathcal{X}^{|A|}$, where $X_A \triangleq \{X_i | i \in A\}$.

A. Graphical Models and Structure Learning

As mentioned before, an undirected graphical model corresponding to a probability distribution is specified by an undirected graph $G = (V, E)$, with each vertex $v_i \in V$ corresponding to a random variable X_i which is a component of a p-dimensional random vector X (for ease of notation, henceforth when we mention a node, we refer to the physical node in the graph, and the associated random variable. The exact meaning should be clear from the context). The edges $E \subseteq V \times V$ of a graphical model can be viewed as encoding the probability distribution of X in several ways, all of which are equivalent under certain conditions. For the purposes of structure learning, an important interpretation is the *local Markov* property, stated below.

Definition 1. ***(Local Markov)*** *Given $G(V, E)$, let $N(i) = \{j \in V | (i, j) \in E\}$ denote the neighborhood of node i. Then a random vector X is said to obey the local Markov property with respect to the graph G if for every $X_i \in V$, conditioned on the nodes in the neighborhood of i, the node i is independent of the remaining nodes in the graph. Mathematically, this means that for any set $B \in V \setminus \{i\} \cup N(i)$, we have that $P(x_i | x_{N(i)}, x_B) = P(x_i | x_{N(i)})$ for all $(x_i, x_{N(i)}, x_B) \in \mathcal{X}^{1+|N(i)|+|B|}$. We henceforth write this as $X_i \overset{X_{N(i)}}{\perp\!\!\!\perp} X_{V \setminus \{i\} \cup N(i)}$.*

Finally, the structure learning problem is stated formally as follows: *given n i.i.d. samples drawn from a random variable X with MRF G, give a learning algorithm and associated conditions such that the hypothesis of the algorithm, $\hat{G}$, is equal to the true MRF G with probability greater than $1 - \delta$.*

B. Conditional Entropy Tests

As we described before in the introduction, a comparison-test based method of structure learning is based on using a test function to compare candidate graphs. Although there are several different implementations, they are all based on the local Markov interpretation of the graph. More specifically, most comparison-test algorithms try to learn the neighborhood of each individual node by comparing potential neighborhoods using a test function. Following the approach of Abbeel et. al. [9], we use conditional entropies as our test function for selecting nodes. In this section, we provide the necessary definitions, and also state some results from information theory that underlies our approach.

First we need to define a few quantities which we use throughout this paper. Given a discrete-valued random variable Y taking values in a finite set $\mathcal{Y}$ such that $\mathbb{P}(Y = y) = p_y \geq 0 \, \forall \, y \in \mathcal{Y}$, and given n i.i.d samples $\{Y^{(i)}\}_{i=1}^n$, the empirical probability mass function $\widehat{P}(y), y \in \mathcal{Y}$ is defined as,

$$\widehat{P}(y) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\{Y^{(i)}=y\}}, \, \forall \, y \in \mathcal{Y}.$$

The empirical entropy $\widehat{H}(Y)$ is defined as the entropy of the empirical distribution $\widehat{P}$.

Next, given two variables Y_1, Y_2, both taking values in $\mathcal{Y}$, we can extend this notation to define empirical conditional measures of the form

$$\widehat{P}(y_1|y_2) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\{Y_1^{(i)}=y_1|Y_2^{(i)}=y_2\}}, \, \forall \, (y_1, y_2) \in \mathcal{Y}^2.$$

Finally, for fixed $y_2 \in \mathcal{Y}$ we define empirical conditional entropy

$$\widehat{H}(Y_1|Y_2 = y_2) = -\sum_{y_1 \in \mathcal{Y}} \widehat{P}(y_1|y_2) \log \widehat{P}(y_1|y_2),$$

and using this we define,

$$\widehat{H}(Y_1|Y_2) = \sum_{y_2 \in \mathcal{Y}} \widehat{P}(y_2) \widehat{H}(Y_1|y_2)$$

Given samples, we use the empirical conditional entropies as given above as the proxy for the actual conditional entropy. Note also that we can define set based versions of all the above statements in a similar manner.

The use of conditional entropies as a test function is motivated by two reasons:

1) By the local Markov property, the conditional entropy for a node is minimized by sets which contain the true neighborhood, and hence (under some weak non-degeneracy conditions), the smallest cardinality set which minimizes the conditional entropy is the true neighborhood.
2) Entropy and measure are related in the sense that two probability measures on a set are close if their entropies are close and vice versa.

The first point is the main reason behind using conditional entropies as a test function, as it reduces the problem of finding a neighborhood to that of finding a set which minimizes an appropriate function, and also indicates a natural greedy sequential approach to selecting the neighbors. We encode this notion in the following proposition, which can be easily derived from the Data Processing Inequality, see [12].

Proposition 1. *For any node $i \in V$, we have that,*

$$H(X_i|X_{N(i)}) \leq H(X_i|X_A),$$

for any set $A \subseteq V \setminus \{i\}$.

The second point can be thought of as indicating that no information is lost if we use entropies instead of measures to learn the structure. This notion can be quantified in terms of the following proposition, which we get by combining Theorem 16.3.2 and Lemma 16.3.1 from [12].

Proposition 2. *Let P and Q be two probability mass functions in a finite set $\mathcal{X}$, with entropies $H(P)$ and $H(Q)$ respectively, and with total variational distance $||P - Q||_1$ given by:*

$$||P - Q||_1 = \sum_{x \in \mathcal{X}} |P(x) - Q(x)|.$$

Then

$$|H(P) - H(Q)| \leq -||P - Q||_1 \log \frac{||P - Q||_1}{|\mathcal{X}|}. \qquad (1)$$

Further, if the relative entropy between them is given by $D(P||Q)$, then

$$D(P||Q) \geq \frac{1}{2 \log 2} ||P - Q||_1^2. \qquad (2)$$

We use this proposition in several places in subsequent proofs. At a high level, (1) allows us to leverage results of convergence of empirical measures to the true measure to obtain similar guarantees on the empirical entropy, while (2) is used to convert entropy conditions to equivalent conditions on the measure (in particular, this allows us to state our non-degeneracy conditions directly in terms of the conditional entropy, instead of more complicated statements in terms of probability distributions usually found in literature [8]).

III. The GreedyAlgorithm(ϵ) Structure Learning Algorithm

In this section, we present our greedy structure learning algorithm, which we henceforth refer to as GreedyAlgorithm(ϵ). We also argue that it always has a low *worst-case* computation complexity, owing to its greedy nature. The challenge however is to find conditions that guarantee correctness, and this question is addressed in subsequent sections.

At a high level, our algorithm considers each node separately, and adds nodes to its neighborhood sequentially in a greedy manner. In particular, at each step we find the node that provides the highest reduction in conditional entropy when added to the existing set. We stop when this reduction is smaller than ϵ.

More specifically, GreedyAlgorithm(ϵ) takes as input the n samples and a single 'threshold' value ϵ. Given any node i, the candidate neighborhood $\widehat{N}(i)$ of the node is initially set to ϕ and is learnt in a sequential manner. In the first stage, the node $j \neq i$ which minimizes the conditional entropy $H(X_i|X_j)$ is chosen as a candidate neighbor, and is added to $\widehat{N}(i)$ if conditioning on the node j reduces the entropy by at-least $\epsilon/2$. In any subsequent stage, a candidate node $k \in V \setminus \widehat{N}(i)$ is chosen as one which minimizes $H(X_i|X_k, H_{\widehat{N}(i)})$, and is added if it reduces the conditional entropy by at-least $\epsilon/2$. At any stage when this condition is not satisfied, the algorithm outputs $\widehat{N}(i)$ and moves on to the next node.

GreedyAlgorithm(ϵ) for structure learning is formally presented in Algorithm 1.

Algorithm 1 GreedyAlgorithm(ϵ)

```
1:  for $i \in V$ do
2:    complete ← FALSE
3:    $\widehat{N}(i) \leftarrow \Phi$
4:    while !complete do
5:      $j = \operatorname{argmin}_{k \in V \setminus \widehat{N}(i)} \widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_k)$
6:      if $\widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_j) < \widehat{H}(X_i \mid X_{\widehat{N}(i)}) - \frac{\epsilon}{2}$ then
7:        $\widehat{N}(i) \leftarrow \widehat{N}(i) \cup \{j\}$
8:      else
9:        complete ← TRUE
10:     end if
11:   end while
12: end for
```

Since the algorithm is greedy, we can characterize its worst case computational complexity independent of its correctness guarantees.

Proposition 3. *The running time of Algorithm 1 is $O(np^4)$ where n is the number of samples and p is the number of random variables.*

Proof: The outer for loop is executed $O(p)$ times. For every iteration of the outer for loop, the $while$ loop (lines 4-11) is run $O(p)$ times. In every iteration of the $while$ loop, line 5 calculates the empirical entropy conditioned on each of the nodes in $\widehat{N}(i)$. Thus, in the worst case, the algorithm performs $O(p^3)$ comparison tests (empirical conditional entropy calculation from samples). Even assuming a naive implementation of a single comparison test that takes $O(np)$, the overall time taken by the algorithm is $O(np^4)$. ■

This shows that GreedyAlgorithm(ϵ) always has low computational complexity for any graph (and in particular, in Section IV, we show that for a large class of graphs, the algorithm has running time of $O(np^2)$). The tradeoff is however in correctness guarantees. The problem arises in the fact that unlike other comparison-test algorithms which are designed to ensure certain correctness guarantees, our algorithm is designed more from the point of view of simplicity and low computational costs. Therefore to derive theoretical guarantees for the algorithm, it is first important to understand the failure mechanism of the algorithm.

IV. Sufficient Conditions for General Discrete Graphical Models

In this section, we provide guarantees for general discrete graphical models, under which GreedyAlgorithm(ϵ) recovers the graphical model structure exactly. First, using an example, we build up intuition for the sufficient conditions, and define two key notions: non-degeneracy conditions and correlation decay. Our main result is presented in Section IV-B, wherein we give a sufficient condition for the correctness of the algorithm in general discrete graphical models.

A. Non-Degeneracy and Correlation Decay

Before analyzing the correctness of structure learning from samples, a simpler problem worth considering is one of algorithm consistency, i.e., does the algorithm succeed to identify the true graph *given the true conditional distributions* (or in other words, given an infinite number of samples). It turns out that the algorithm as presented in Algorithm 1 does not even possess this property, as is illustrated by the following counter-example

Let $V = \{0, 1, \cdots, D, D+1\}$, $X_i \in \{-1, 1\} \forall i \in V$ and $E = \{\{0, i\}, \{i, D+1\} \mid 1 \leq i \leq D\}$. Let $P(x_V) = \frac{1}{Z} \prod_{\{i,j\} \in E} e^{\theta x_i x_j}$, where Z is a normalizing constant (this is the classical zero-field Ising model potential). The graph is shown in Fig. 1.

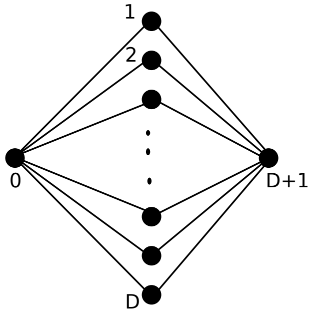

Fig. 1. An example of adding spurious nodes: Execution of GreedyAlgorithm(ϵ) for node 0 adds node $D+1$ in the first iteration, even though it is not a neighbor.

Suppose the actual entropies are given as input to Algorithm 1. It can be shown in this case that for a given θ, there exists a D_{thresh} such that if $D > D_{\text{thresh}}$, then the output of Algorithm 1 will select the edge $\{0, D+1\}$ in the first iteration. This is easily understood because if D is large, the distribution of node 0 is best accounted for by node $D+1$, although it is not a neighbor. Thus, even with exact entropies, the algorithm will always include edge $(0, D+1)$, although it does not exist in the graph.

The algorithm can however easily be shown to satisfy the following weaker consistency guarantee: given infinite samples, for any node in the graph, the algorithm will return a *super-neighborhood*, i.e., a superset of the neighborhood of i. This suggests a simple fix to obtain a consistent algorithm, as we can follow the greedy phase by a 'node-pruning' phase, wherein we test each node in the neighborhood of a node i returned by the algorithm (to do this, we can compare the entropy of i conditioned on the neighborhood with and without a node, and remove it if they are the same). However the problem is complicated by the presence of samples, as pruning a large super-neighborhood requires calculating estimates of entropy conditioned on a large number of nodes, and hence this drives up the sample complexity. In the rest of the paper, we avoid this problem by ignoring the pruning step, and instead prove a stronger correctness guarantee: given any node i, the algorithm always picks a *correct* neighbor of i as long as any one remains undiscovered. Towards this end, we first define two conditions which we require for the correctness of GreedyAlgorithm(ϵ).

Assumption 1 (Non-degeneracy). *Choose a node i. Let $N(i)$ be the set of its neighbors. Then $\exists \epsilon > 0$ such that $\forall\ A \subset N(i)$, $\forall\ j \in N(i) \setminus A$ and $\forall\ l \in N(j) \setminus \{i\}$, we have that*

$$H(X_i \mid X_A) - H(X_i \mid X_A, X_j) > \epsilon \text{ and} \tag{3}$$

$$H(X_i \mid X_A, X_l) - H(X_i \mid X_A, X_j, X_l) > \epsilon \tag{4}$$

Assumption 1 is illustrated in Fig. 2.

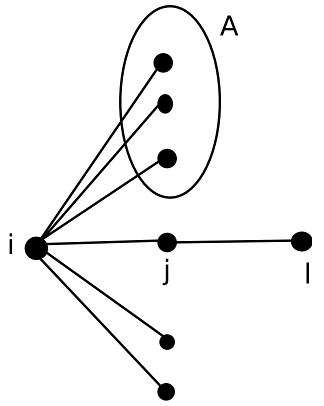

Fig. 2. Non-degeneracy condition for node i: (i) Entropy of i conditioned on any sub-neighborhood A reduces by at-least ϵ if any other neighbor j is added to the conditioning set, (ii) Entropy of i conditioned on A and a two hop neighbor l reduces by at-least ϵ if the corresponding one hop neighbor j is added to the conditioning set

Assumption 2 (Correlation Decay). *Choose a node i. Let $N^1(i)$ and $N^2(i)$ be the sets of its 1-hop and 2-hop neighbors respectively. Choose another set of nodes B. Let $d(i, B) = \min_{j \in B} d(i, j)$, where $d(i, j)$ denotes the distance between nodes i and j. Then, we have that $\forall x_i, x_{N^1(i)}, x_{N^2(i)}, x_B$*

$$\left|P(x_i, x_{N^1(i)}, x_{N^2(i)} \mid x_B) - P(x_i, x_{N^1(i)}, x_{N^2(i)})\right| < f(d(i, B))$$

where f is a monotonic decreasing function.

Assumption 1 (or a variant thereof) is a standard assumption for showing correctness of any structure learning algorithm, as it ensures that there is a *unique* minimal graphical model for the distribution from which the samples are generated. Although the way we state the assumption is tailored to our algorithm, it can be shown to be equivalent to similar assumptions in literature [8]. Informally speaking, Assumption 1 states that for node i, any 2-hop neighbor captures less information about node i than the corresponding 1-hop neighbor. In the case of a Markov Chain, Assumption 1 reduces to a weaker version of an $\epsilon-$Data Processing Inequality (i.e., DPI with an epsilon gap), and in a sense, Assumption 1 can be viewed as a generalized $\epsilon-$DPI for networks with cycles.

On the other hand, Assumption 2 along with large girth implies that the information a node j has about node i is 'almost Markov' along the shortest path between i and j. This in conjunction with Assumption 1 implies that for any two nodes i and k, the information about i captured by k is less than that captured by j where j is the neighbor of i on the shortest path between i and k.

B. Guarantees for the Recovery of a General Graphical Model

We now state our main theorem, wherein we give a sufficient condition for correctness of GreedyAlgorithm(ϵ) in a general graphical model.

The counter-example given in Section IV-A suggests that the addition of spurious nodes to the neighborhood of i is related to the existence of non-neighboring nodes of i which somehow accumulate sufficient influence over it. The accumulation of influence is due to slow decay of influence on short paths (corresponding to a high θ in the example), and the effect of a large number of such paths (corresponding to high D). Correlation decay (Assumption 2) allows us to control the first. Intuitively, the second can be controlled if the neighborhood of i is 'locally tree-like'. To quantify this notion, we define the girth of a graph Girth(G) to be the length of the smallest cycle in the graph G. Now we have the following theorem.

Theorem 1. *Consider a graphical model G where the random variable corresponding to each node takes values in a set $\mathcal{X}$ and satisfies the following:*

- *Non-degeneracy (Assumption 1) with parameter ϵ,*
- *Correlation decay (Assumption 2) with decay function $f(\cdot)$,*
- *Maximum degree D.*

Define $h \triangleq h(\epsilon, D) \triangleq \frac{\epsilon^2 |\mathcal{X}|^{-(D+1)^4}}{256}$ and suppose $f^{-1}(h)$ exists. Further suppose G obeys the following condition:

$$\textit{Girth}(G) \triangleq g > 2\left(f^{-1}(h) + 1\right). \tag{5}$$

Then, given $\delta > 0$, GreedyAlgorithm(ϵ) recovers G exactly with probability greater than $1 - \delta$ with sample complexity $n = \xi\left(\epsilon^{-4} \log \frac{p}{\delta}\right)$, where ξ is a constant independent of p, ϵ and δ.

The proof follows from the following two lemmas. Lemma 1 implies that if we had access to actual entropies, Algorithm 1 always recovers the neighborhood of a node exactly. Lemma 2 shows that with the number of samples n as stated in Theorem 1, the empirical entropies are very close to the actual entropies with high probability and hence Algorithm 1 recovers the graphical model structure exactly with high probability even with empirical entropies.

Lemma 1. *Consider a graphical model G in which node i satisfies Assumptions 1 and 2. Let the girth of the graph be $g > 2\left(f^{-1}(h) + 1\right)$. Then, $\forall\ A \subset N(i)$, $u \notin N(i)$, $\exists\ j \in N(i) \setminus A$ such that*

$$H(X_i \mid X_A, X_j) < H(X_i \mid X_A, X_u) - \epsilon + \widehat{\epsilon} \tag{6}$$

where $\widehat{\epsilon} = |\mathcal{X}|^{(D+1)^2} \sqrt{h}$, and h is as defined in Theorem 1.

Proof: If A separates i and u in G then $P(x_i|x_A, x_u) = P(x_i|x_A)$ and hence $H(X_i \mid X_A, X_u) = H(X_i \mid X_A)$. Then, the statement of the lemma follows from (3).

Now suppose A does not separate i and u in G. Then, $\exists j \in N(i) \setminus A$ and $l \in N(j) \setminus \{i\}$ such that the shortest path between i and u in the induced sub graph on $V \setminus A$ passes through j and l. Assumption 1 implies that $H(X_i \mid X_A, X_l) - H(X_i \mid X_A, X_j, X_l) > \epsilon$. Now, choose a subset B of nodes such that $A \cup B \cup \{j\}$ separates i and l in graph G and $d(i, B) \geq \frac{g-2}{2}$, where g is the girth of the graph. Note that such a B (possibly empty) exists since the girth of the graph is g. From Assumption 2, we know that

$$\begin{aligned}
&|P(x_i, x_{N(i)\cup N^2(i)}) - P(x_i, x_{N(i)\cup N^2(i)} \mid x_B)| < f\left(\tfrac{g}{2} - 1\right)\\
&\Rightarrow \sum_{x_i, x_A, x_j} |P(x_i, x_A, x_j) - P(x_i, x_A, x_j \mid x_B)|\\
&\qquad\qquad < |\mathcal{X}|^{(D+1)^2} f\left(\tfrac{g}{2} - 1\right) \;\; \forall\, x_B\\
&\Rightarrow H(X_i, X_A, X_j) - H(X_i, X_A, X_j \mid X_B)\\
&\qquad < -|\mathcal{X}|^{(D+1)^2} f\left(\tfrac{g}{2} - 1\right)\left(\log f\left(\tfrac{g}{2} - 1\right)\right) \triangleq \widehat{\epsilon}\\
&\Rightarrow \left(H(X_i \mid X_A, X_j) + H(X_A, X_j)\right) -\\
&\quad \left(H(X_i \mid X_A, X_j, X_B) + H(X_A, X_j \mid X_B)\right) < \widehat{\epsilon}\\
&\Rightarrow H(X_i \mid X_A, X_j) - H(X_i \mid X_A, X_j, X_B) < \widehat{\epsilon},
\end{aligned}$$

where the first implication follows from marginalizing irrelevant variables and the second implication follows from (1). Using this we have that,

$$\begin{aligned}
&H(X_i \mid X_A, X_j, X_l)\\
&\;\geq H(X_i \mid X_A, X_j, X_l, X_B)\\
&\;= H(X_i \mid X_A, X_j, X_B) \text{ since } X_i \overset{X_A, X_j, X_B}{\perp\!\!\!\perp} X_l\\
&\;> H(X_i \mid X_A, X_j) - \widehat{\epsilon}
\end{aligned}$$

Using a similar argument, we also have,

$$H(X_i \mid X_A, X_l, X_u) > H(X_i \mid X_A, X_l) - \widehat{\epsilon}$$

Combining the two inequalities, and using the fact that under the given conditions $\widehat{\epsilon} < \frac{\epsilon}{8}$, we get

$$H(X_i \mid X_A, X_j) \leq H(X_i \mid X_A, X_u) - \frac{3\epsilon}{4}.$$

■

Lemma 2. *Consider a graphical model G in which each node takes values in $\mathcal{X}$. Let the girth of the graph be $g > 2\left(f^{-1}(h) + 1\right)$ and the number of samples be $n > 2^{18}\epsilon^{-4}\left((D+2)\log 2|\mathcal{X}| + \log \frac{p}{\delta}\right)$. Let $\widehat{P}$ and $\widehat{H}$ denote the empirical probability and empirical entropy as defined in Section II-B.*

Then $\forall\, i \in G$ such that Assumptions 1 and 2 are satisfied, with probability greater than $1 - \frac{\delta}{p}$, we have that $\forall\, A \subsetneq N(i)$, $u \notin N(i)$, $\exists\, j \in N(i) \setminus A$ such that

$$\widehat{H}(X_i \mid X_A, X_j) < \widehat{H}(X_i \mid X_A, X_u) - \frac{\epsilon}{2} \qquad (7)$$

and $\forall\, i$, $A \subset N(i)$, $j \in N(i) \setminus A$, we have that

$$\widehat{H}(X_i \mid X_A, X_j) < \widehat{H}(X_i \mid X_A) - \frac{\epsilon}{2} \qquad (8)$$

Due to lack of space, we only provide a proof outline. The complete proof can be found in [13].

First, we use the fact that given sufficient samples, the empirical measure is close to the true measure uniformly in probability. Specifically, given any subset $A \subseteq V$ of nodes and any fixed $x_A \in \mathcal{X}^{|A|}$, we have by Azuma's inequality after n samples,

$$\mathbb{P}\left[\left|P(x_A) - \widehat{P}(x_A)\right| > \gamma\right] < 2e^{-2\gamma^2 n}.$$

The crux of the proof lies in the fact that given the above sample complexity, this statement holds uniformly over all the sets we are interested in (by union bound). Finally using Proposition 2, we can translate this statement to conditional entropies.

Proof (Theorem 1): The proof is based on mathematical induction. The induction claim is as follows: just before entering an iteration of the WHILE loop, $\widehat{N}(i) \subset N(i)$. Clearly this is true at the start of the WHILE loop since $\widehat{N}(i) = \Phi$. Suppose it is true just after entering the k^{th} iteration. If $\widehat{N}(i) = N(i)$ then clearly $\forall j \in V \setminus \widehat{N}(i)$, $H(X_i \mid X_{\widehat{N}(i)}, X_j) = H(X_i \mid X_{\widehat{N}(i)})$. Since with probability greater than $1 - \frac{\delta}{p}$, $\left|\widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_j) - H(X_i \mid X_{\widehat{N}(i)}, X_j)\right| < \frac{\epsilon}{8}$ and $\left|\widehat{H}(X_i \mid X_{\widehat{N}(i)}) - H(X_i \mid X_{\widehat{N}(i)})\right| < \frac{\epsilon}{8}$, we have that $\left|\widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_j) - \widehat{H}(X_i \mid X_{\widehat{N}(i)})\right| < \frac{\epsilon}{4}$. So control exits the loop without changing $\widehat{N}(i)$. On the other hand, if $\exists j \in N(i) \setminus \widehat{N}(i)$, then from (8) of Lemma 2, we have that $\widehat{H}(X_i \mid X_{\widehat{N}(i)}) - \widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_j) > \frac{\epsilon}{2}$. So, a node is chosen to be added to $\widehat{N}(i)$ and control does not exit the loop. Now suppose for contradiction that a node $u \notin N(i)$ is added to $\widehat{N}(i)$. Then we have that $\widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_u) < \widehat{H}(X_i \mid X_{\widehat{N}(i)}, X_j)$. But this contradicts (7) from Lemma 2. Thus, a neighbor $j \in N(i) \setminus \widehat{N}(i)$ is picked in the iteration to be added to $\widehat{N}(i)$, proving that the neighborhood of i is recovered exactly with probability greater than $1 - \frac{\delta}{p}$. Using union bound, it is easy to see that the neighborhood of each node (i.e., the graph structure) is recovered exactly with probability greater than $1 - \delta$. ■

Remark 1. *The proof for Theorem 1 can also be used to provide node-wise guarantees, i.e., for every node satisfying Assumptions 1 and 2, if the number of samples is sufficiently large (in terms of its degree, and the length of the smallest cycle it is part of), its neighborhood will be recovered exactly with high probability.*

Remark 2. *Any decreasing correlation-decay function f suffices for Theorem 1. However, the faster the correlation decay, the smaller the girth in the sufficient condition for Theorem 1 needs to be.*

And finally we have a corollary for the computational complexity of GreedyAlgorithm(ϵ).

Corollary 1. *The expected run time of Algorithm 1 is $O\left(\delta n p^4 + (1-\delta) D n p^2\right)$. Further, if δ is chosen to be*

$O(p^{-2})$, the sample complexity n is $O(\log p)$ and the expected run time of Algorithm 1 is $O(np^2 \log p)$.

Proof: For the second part, note that with probability greater than $1-\delta$, the algorithm recovers the correct graph structure exactly. In this case, the number of iterations of the *while* loop is bounded by D for each node and hence the total run time is $O(Dnp^2)$. Using the previous worst case bound on the running time, we get the result. ■

V. Guarantees for the Recovery of an Ising Graphical Model

In this section, we show how Theorem 1 can be used to efficiently learn Ising graphical models satisfying certain conditions. The zero field Ising model is a pairwise, symmetric, binary graphical model which is widely used in statistical physics to model the alignment of magnetic spins in a magnetic field. It is defined as follows:

Definition 2. *A set of random variables $\{X_v \mid v \in V\}$ are said to be distributed according to a zero field Ising model if*

1) $X_v \in \{-1,1\}\ \forall v \in V$ *and*
2) $P(x_V) = \frac{1}{Z} \prod_{i,j \in V} \exp(\theta_{ij} x_i x_j)$

where Z is a normalizing constant. The graphical model of such a set of random variables is given by $G(V,E)$ where $E = \{\{i,j\} \mid \theta_{ij} \neq 0\}$.

It is easy to verify that this satisfies the local Markov property. Another very useful property of zero-field Ising models is that they are symmetric with respect to -1 and 1. Formally, if P is the probability distribution function over a set of zero-field Ising distributed random variables, then, $P(x_V) = P(-x_V)$.

The main contribution of this section is in the form of the following theorem, which translates the sufficient conditions from Section IV to equivalent conditions for an Ising model.

Theorem 2. *Consider a zero-field Ising model on a graph G with maximum degree D. Let the edge parameters θ_{ij} be bounded in the absolute value by $0 < \beta < |\theta_{ij}| < \frac{\log 2}{2D}$. Let $\epsilon \triangleq 2^{-7} e^{-6\gamma D} \sinh^2(2\beta)$. If the girth of the graph satisfies $g > \frac{2^{15}}{\log 2}\left\{D^2 \log 2 - \log\left(\sinh 2\beta\right)\right\}$ then with samples $n = \xi \epsilon^{-4} \log \frac{p}{\delta}$ (where ξ is a constant independent of ϵ, δ, p), GreedyAlgorithm(ϵ) outputs the exact structure of G with probability greater than $1-\delta$.*

The proof of this theorem consists of showing that an Ising graphical model satisfies Assumptions 1 and 2 if the graph has large girth and the parameters on the edges satisfy certain conditions. In Section V-A, we show that under certain conditions, an Ising model has an almost exponential correlation decay. Then in Section V-B, we use the correlation decay of Ising models to show that under some further conditions, they also satisfy Assumption 1 for non-degeneracy. Combining the two, we get the above sufficient conditions for GreedyAlgorithm(ϵ) to learn the structure of an Ising graphical model with high probability.

A. Correlation Decay in Ising Models

We will start by proving the validity of Assumption 2 in the form of the following proposition.

Proposition 4. *Consider a zero-field Ising model on a graph G with maximum degree D. Let the edge parameters θ_{ij} be bounded in the absolute value by $0 < \beta < |\theta_{ij}| < \gamma$ where $\beta < \gamma < \frac{\log 2}{2D}$. Then, for any node i, its neighbors $N^1(i)$, its 2-hop neighbors $N^2(i)$ and a set of nodes A, we have*

$$\left|P(x_i, x_{N^1(i)}, x_{N^2(i)} \mid x_A) - P(x_i, x_{N^1(i)}, x_{N^2(i)})\right| < c \exp\left(-\frac{\log 2}{3} \min\left(d(i,A), \frac{g-1}{2}\right)\right)$$

$\forall\ x_i, x_{N^1(i)}, x_{N^2(i)}$ and x_A (where c is a constant independent of i and A).

We give an overview of the proof, quoting the necessary lemmas as we go. The proofs of the lemmas are omitted due to lack of space, but can be found in [13].

The outline of the proof of Proposition 4 is as follows. First, we show that if a subset of nodes is conditioned on a Markov blanket (i.e., on another subset of nodes which separates them from the remaining graph), then their potentials remain the same. For this we have the following lemma.

Lemma 3. *Consider a graphical model $G(V,E)$ and the corresponding factorizable probability distribution function P. Let A, B and C be a partition of V and B separate A and C in G. Let $\tilde{G}(A \cup B, \tilde{E})$ be the induced subgraph of G on $A \cup B$, with the same edge potentials as G on all its edges and $\tilde{P}$ be the corresponding probability distribution function. Then, we have that $P(x_D \mid x_B) = \tilde{P}(x_D \mid x_B)\ \forall\ x_D, x_B$ where $D \subseteq A$.*

Now, for any node i, the induced subgraph on all nodes which are at distance less than $\frac{g}{2} - 1$ is a tree. Thus we can concentrate on proving correlation decay for a tree Ising model. We do this through the following steps:

1) Without loss of generality, the tree Ising model can be assumed to have all positive edge parameters
2) The worst case configuration for the conditional probability of the root node is when all the leaf nodes are set to the same value and all the edge parameters are set to the maximum possible value
3) For this scenario, correlation decays exponentially

The following three lemmas encode these three steps.

Lemma 4. *Consider a tree Ising graphical model T. Let the corresponding probability distribution be P. Replace all the edge parameters on this graphical model by their absolute values. Let the corresponding probability distribution after this change be $\tilde{P}$. Then, there exists a set of bijections $\{M_v : \{-1,1\} \to \{-1,1\} \mid v \in V \setminus \{r\}\}$ where V is the set of vertices and r is the root node such that, $\forall x_r, x_{V \setminus r}$ we have that $P(x_r, x_{V \setminus r}) = \tilde{P}(x_r, M_v(x_v), v \in V \setminus r)$.*

Lemma 5. *For a tree Ising graphical model T with root r and set of leaves L, we have*

$$(x_r = 1, x_L = 1) \in \arg\max_{x_r, x_L} |P(x_r \mid x_L) - P(x_r)|$$

And finally we have the following lemma.

Lemma 6. *In a tree Ising model, suppose $|\theta_{ij}| < \gamma < \frac{\log 2}{2D}$ where D is the maximum degree of the graph. Then we have exponential correlation decay between a node i, its neighbors $N^1(i)$, its 2-hop neighbors $N^2(i)$ and the set of leaves L i.e., $\left|\tilde{P}(x_i, x_{N^1(i)}, x_{N^2(i)} \mid x_L) - \tilde{P}(x_i, x_{N^1(i)}, x_{N^2(i)})\right| < c\exp(-\frac{\log 2}{3} d(i, L))$ where c is a constant independent of the nodes considered.*

B. Non-degeneracy in Ising Models with Correlation Decay

Now using the results from the previous section, we turn our attention to the question of correlation decay. In particular, we have the following lemma which says that if an Ising graphical model has almost exponential correlation decay and its edge parameters satisfy certain conditions, then it also satisfies Assumption 1. For the proof, refer to [13].

Lemma 7. *Consider an Ising graphical model with potentials θ_{ij} bounded by $0 < \beta < |\theta_{ij}| < \gamma$, max degree D, and having correlation decay as follows*

$$\begin{aligned} &\left|P(x_i, x_{N^1(i)}, x_{N^2(i)}) - P(x_i, x_{N^1(i)}, x_{N^2(i)} | x_B)\right| \\ &\quad < c \exp\left(-\alpha \min\left(d(i, B), \tfrac{g-2}{2}\right)\right) \end{aligned}$$

$\forall\ i, B, x_i, x_{N^1(i)}, x_{N^2(i)}$. If the girth $g > \frac{2}{\alpha}\Big\{(D + 8)\log 2 + \log c + \log\left(1 + 2^D e^{2\gamma}\right) + 2\gamma(D + 2) - \log\left(\sinh 2\beta\right)\Big\}$, then this graphical model satisfies Assumption 1 with $\epsilon = 2^{-7} e^{-6\gamma D} \sinh^2(2\beta)$.

Finally, the proof of Theorem 2 follows directly by combining Theorems 1 and 4 and Lemma 7. For complete details, refer to [13].

VI. Discussion

We developed a simple greedy algorithm for Markov structure learning. The algorithm is simple to implement and has low computational complexity. We then showed that under some non-degeneracy, correlation decay, maximum degree and girth assumptions on the MRF, our algorithm recovers the correct graph structure with $O(\epsilon^{-4} \log \frac{p}{\delta})$ samples. We then specialize our conditions to prove a self-contained result for the most popular discrete graphical model - the Ising model.

The success of our algorithm can be further improved by post-processing via *pruning*. In particular, as mentioned, the neighborhood of a node as estimated by our algorithm always includes the true neighborhood – but it may also include spurious nodes. The latter can be then identified by checking each node of the estimated neighborhood, to see if it actually provide a reduction in conditional entropy over and above all the other nodes. Analysis of the improvement achieved by such a procedure is more challenging, but it may be likely that doing so will reveal an algorithm that can handle much larger degrees and smaller girths.

Acknowledgements

This work was partially supported by ARO grant W911NF-10-1-0360.

References

[1] N. P. Santhanam and M. J. Wainwright, "Information-theoretic limits of selecting binary graphical models in high dimensions," *CoRR*, vol. abs/0905.2639, 2009.

[2] J. Bento and A. Montanari, "Which graphical models are difficult to learn?," 2009. http://arxiv.org/abs/0910.5761.

[3] N. Srebro, "Maximum likelihood bounded tree-width markov networks," *Artificial Intelligence*, vol. 143, no. 1, pp. 123 – 138, 2003.

[4] A. Bogdanov, E. Mossel, and S. P. Vadhan, "The complexity of distinguishing markov random fields," in *APPROX-RANDOM*, pp. 331–342, 2008.

[5] M. J. Wainwright and M. I. Jordan, *Graphical Models, Exponential Families, and Variational Inference*. Hanover, MA, USA: Now Publishers Inc., 2008.

[6] P. Ravikumar, M. J. Wainwright, and J. D. Lafferty, "High-dimensional graphical model selection using l_1-regularized logistic regression," *Annals of Statistics*, vol. 38, no. 3, pp. 1287–1319, 2010.

[7] O. Banerjee, L. El Ghaoui, and A. d'Aspremont, "Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data," *J. Mach. Learn. Res.*, vol. 9, pp. 485–516, 2008.

[8] G. Bresler, E. Mossel, and A. Sly, "Reconstruction of markov random fields from samples: Some observations and algorithms," in *APPROX '08 / RANDOM '08*, (Berlin, Heidelberg), pp. 343–356, Springer-Verlag, 2008.

[9] P. Abbeel, D. Koller, and A. Y. Ng, "Learning factor graphs in polynomial time and sample complexity," *J. Mach. Learn. Res.*, vol. 7, pp. 1743–1788, 2006.

[10] C. I. Chow and C. N. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Transactions on Information Theory*, vol. 14, pp. 462–467, 1968.

[11] V. Y. F. Tan, A. Anandkumar, and A. S. Willsky, "Learning high-dimensional markov forest distributions: Analysis of error rates," *CoRR*, vol. abs/1005.0766, 2010.

[12] T. M. Cover and J. A. Thomas, *Elements of Information Theory, 1st Edition (Wiley Series in Telecommunications and Signal Processing)*. New York, NY, USA: Wiley-Interscience, 2006.

[13] P. Netrapalli, S. Banerjee, S. Sanghavi, and S. Shakkottai, "Greedy learning of markov network structure," tech. rep., The University of Texas at Austin, 2010.