Submitted to the Annals of Statistics
arXiv: stat.ML/1203.3887

LEARNING LOOPY GRAPHICAL MODELS WITH LATENT VARIABLES: EFFICIENT METHODS AND GUARANTEES*

By Animashree Anandkumar† and Ragupathyraj Valluvan†

Univ. of California Irvine

The problem of structure estimation in graphical models with latent variables is considered. We characterize conditions for tractable graph estimation and develop efficient methods with provable guarantees. We consider models where the underlying Markov graph is locally tree-like and the model is in the regime of correlation decay. For the special case of the Ising model, the number of samples n required for structural consistency of our method scales as $n = \Omega(\theta_{\min}^{-\delta\eta(\eta+1)-2} \log p)$, where p is the number of variables, $\theta_{\min}$ is the minimum edge potential, δ is the depth (i.e., distance from a hidden node to the nearest observed nodes), and η is a parameter which depends on the bounds on node and edge potentials in the Ising model. Necessary conditions for structural consistency under any algorithm are derived and our method nearly matches the lower bound on sample requirements. Further, the proposed method is practical to implement and provides flexibility to control the number of latent variables and the cycle lengths in the output graph.

1. Introduction. Learning latent variable models from observed samples involves mainly two tasks: discovering relationships between the observed and hidden variables, and estimating the strength of such relationships. One of the simplest latent variable models is the so-called *latent class model* or *näive Bayes model*, where the observed variables are conditionally independent given the state of the latent factor. An extension of these models are *latent tree models* with many hidden variables forming a tree hierarchy. Latent tree models have been effective in modeling data in a variety of domains, such as the evolutionary process which gave rise to the present-day species in bio-informatics (popularly known as *phylogenetics*) [20, 42], for financial and topic modeling [16], and for modeling contextual information for object recognition in computer vision [15]. Prior works on learning latent tree models (e.g. [16, 22, 34]), demonstrate that latent tree models can be learnt efficiently in high dimensions. In other words, the number of samples required for consistent learning is much smaller than the number of variables at hand. Moreover, inference in latent tree models is computationally tractable by means of simple algorithms such as *belief propagation*.

Despite all the above advantages, the assumption of a tree structure may be too restrictive. For instance, in an analysis of the relationships between topics (encoded as latent variables) and words (corresponding to observed variables), a latent tree model posits that the words are generated from a single topic, while, in reality there are common words across topics. Loopy graphical models are able to capture such relationships, while retaining many advantages of the latent tree models.

Relaxing the tree assumption leads to non-trivial challenges: in general, learning these models is NP-hard [7, 27], even when there are no latent variables, and developing methods for learning

*An abridged version of this paper appears in Proc. of NIPS 2012.

†The first author is supported in part by the NSF Award CCF-1219234, AFOSR Award FA9550-10-1-0310, and ARO Award W911NF-12-1-0404. The second author is supported by the ONR award N00014-08-1-1015.

AMS 2000 subject classifications: Primary 62H12; secondary 05C12

Keywords and phrases: Graphical model selection, Latent variables, Quartet methods

such fully observed models is itself an area of active research (e.g. [3, 26, 39]). In this paper, we consider structure estimation in latent graphical models Markov on *locally tree-like* graphs, meaning that local neighborhoods in the graph do not contain cycles. Learning such graphs has many non-trivial challenges: are there parameters regimes where these models can be learnt consistently and efficiently? If so, are there practical learning algorithms? Are learning guarantees for loopy models comparable to those for latent trees? How does learning depend on various graph attributes such as node degrees, girth of the graph, and so on? We provide answers to these questions in this paper.

1.1. *Our Approach and Contributions.* We consider learning latent graphical models Markov on locally tree-like graphs in the regime of correlation decay. In this regime, there are no long-range correlations, and the local statistics converge to a tree limit. The implication of correlation decay is immediately clear: we can employ the available latent tree methods to learn "local" subgraphs consistently, as long as they do not contain any cycles. However, a non-trivial challenge remains: how does one merge these estimated local subgraphs (i.e., latent trees) to obtain an overall graph estimate? Specifically, merging involves matching latent nodes across different latent tree estimates, and it is not clear if this can be performed in an efficient manner.

We employ a different philosophy for building locally tree-like graphs with latent variables. We decouple the process of introducing cycles and latent variables in the output model. We initialize a loopy graph consisting of only the observed variables, and then iteratively add latent variables to local neighborhoods of the graph. We establish correctness of our method under a set of natural conditions.

We provide precise conditions for structural consistency of LocalCLGrouping under the probably approximately correct (PAC) model of learning [28, p. 7] for general discrete models. We simplify these conditions for the Ising model, where each node is a binary random variable, to obtain better intuitions. We establish that for structural consistency, the number of samples is required to scale as $n = \Omega(\theta_{\min}^{-\delta\eta(\eta+1)-2} \log p)$, where p is the number of observed variables, $\theta_{\min}$ is the minimum edge potential, δ is the depth (i.e., graph distance from a hidden node to the nearest observed nodes), and η is a parameter which depends on the minimum and maximum node and edge potentials of the Ising model ($\eta = 1$ for homogeneous models). When there are no hidden variables ($\delta = 1$), the sample complexity is strengthened to $n = \Omega(\theta_{\min}^{-2} \log p)$, which matches with the best known sample complexity for learning fully-observed Ising models [3, 26].

We also establish necessary conditions for any (deterministic) algorithm to recover the graph structure, and establish that $n = \Omega(\Delta_{\min}\rho^{-1} \log p)$ samples are necessary for structural consistency, where $\Delta_{\min}$ is the minimum degree and ρ is the fraction of observed nodes. This is comparable to the requirement of the proposed method under uniform node sampling (i.e., selecting the observed nodes uniformly), given by $n = \Omega(\Delta_{\max}^2\rho^{-2}(\log p)^3)$, where $\Delta_{\max}$ is the maximum degree in the graph. Thus, our method is competitive with respect to the lower bound on learning.

Our proposed method has a number of attractive features for practical implementation: the method is amenable to parallelization which makes it efficient on large datasets. The method provides flexibility to control the length of cycles and the number of latent variables introduced in the output model. The method can incorporate penalty scores such as the Bayesian information criterion (BIC) [41] to tradeoff model complexity and fidelity. Moreover, by controlling the cycle lengths in the output model, we can obtain models with good inference accuracy under simple algorithms such as loopy belief propagation (LBP). Preliminary experiments on the newsgroup dataset suggests that the method can discover intuitive relationships efficiently, and also compares well with the popular latent Dirichlet allocation (LDA) [6] in terms of topic coherence and perplexity.

1.2. *Related Work.* The classical *latent class models* (LCM) consists of multivariate distributions with a single latent variable and the observed variables are conditionally independent under each state of the latent variable [31]. Hierarchical latent class (HLC) models [14, 46, 47] generalize these models by allowing multiple latent variables. However, the proposed learning algorithms are based on greedy local search in a high-dimensional space, which is computationally expensive. Moreover, the algorithms do not have theoretical guarantees. Similar shortcomings also hold for expectation-maximization (EM) based approaches [21, 29]. Learning latent trees has been studied extensively before, mainly in the context of phylogenetics. See [20, 42] for a thorough overview. Efficient algorithms with provable performance guarantees are available (e.g. [1, 16, 18, 22]). Our proposed method in this paper is inspired by [16].

Works on high-dimensional graphical model selection are more recent. The approaches can be mainly classified into two groups: local approaches [3, 8, 26, 36] and those based on convex optimization [13, 32, 39, 40]. There is a general agreement that the success of these methods is related to the presence of correlation decay in the model [3, 5]. This work makes the connection explicit: it relates the extent of correlation decay (i.e., the convergence rate to the tree limit) with the learning efficiency for latent models on large girth graphs. An analogous study of the effect of correlation decay for learning fully observed models is presented in [3].

This paper is the first work to provide provable guarantees for learning discrete latent models on loopy graphs in high dimensions (which can also be easily be extended to Gaussian models, see remarks following Theorem 2). The work in [12] considers learning latent Gaussian graphical models using a convex relaxation method. However, the method cannot be easily extended to discrete models. Moreover, the "incoherence" conditions required for the success of convex methods are hard to interpret and verify in general. In contrast, our conditions for success are transparent and based on the presence of correlation decay in the model. The work in [8] considers graphical model selection with hidden variables, but proposes learning Markov graph of marginal distribution (upon marginalizing the hidden variables) and then replacing the cliques in the estimated graphs with hidden variables. Sample complexity results are not provided, and the method performs poorly in high dimensions, since it aims to estimate dense graphs.

2. System Model.

2.1. *Graphical Models.* A *graphical model* is a family of multivariate distributions which are Markov in accordance to a particular undirected graph $G = (W, E)$ [30, p. 32]. For any distribution belonging to the model class, a random variable X_i taking value in a set $\mathcal{X}$ is associated with each node $i \in W$ in the graph. We consider discrete graphical models where $\mathcal{X}$ is a finite set. The set of edges E captures the set of conditional independence relations among the random variables. We say that a set of random variables $\mathbf{X}_W := \{X_i, i \in W\}$ with probability mass function (pmf) P is Markov on the graph G if it factorizes according to the cliques of G:

$$P(\mathbf{x}) = \exp\left(\sum_{c \in \mathcal{C}} \theta_c(\mathbf{x}_c) - A(\boldsymbol{\theta})\right), \quad \forall \mathbf{x} \in \mathcal{X}^m, \tag{1}$$

where $\mathcal{C}$ is the set of cliques of G, $m := |W|$ is the number of variables, and $\mathbf{x}_c$ is the set of configurations corresponding to clique c. The quantity $A(\boldsymbol{\theta})$ is known as the *log-partition function* and serves to normalize the probability distribution. The functions θ_c are known as *potential* functions and correspond to the *canonical* parameters of the exponential family.

A special case is the Ising model, which is the class of pairwise distributions over binary variables $\{-1,+1\}^m$ with probability mass function (pmf) of the form

$$P(\mathbf{x}) = \exp\left(\sum_{e\in E}\theta_{i,j}x_ix_j + \sum_{i\in V}\phi_i x_i - A(\boldsymbol{\theta})\right), \quad \forall \mathbf{x}\in\{-1,1\}^m. \tag{2}$$

We specialize some of our results to the class of Ising models.

We consider a multivariate distribution belonging to the class of latent graphical models in which a subset of nodes is latent or hidden. Let $H \subset W$ denote the hidden nodes and $V = W \setminus H$ denote the observed nodes. Our goal is to discover the presence of hidden variables $\mathbf{X}_H$ and learn the unknown graph structure $G(W)$, given n i.i.d. samples from observed variables $\mathbf{X}_V$. Let $p := |V|$ denote the number of observed nodes and $m := |W|$ denote the total number of nodes.

2.2. *Tractable Graph Families: Girth-Constrained Graphs.* In general, structure estimation of graphical models is NP-hard [7, 27]. We now characterize a tractable class of models for which we can provide guarantees on graph estimation.

We consider the family of graphs with a bound on the *girth*, which is the length of the shortest cycle in the graph. There are many graph constructions which lead to a bound on girth. For example, the bipartite Ramanujan graph [17, p. 107] and the random Cayley graphs [24] have bounds on the girth. Recently, efficient algorithms have been proposed to generate large girth graphs efficiently [4].

Although girth-constrained graphs are locally tree-like, in general, their global structure makes them hard instances for learning. Specifically, girth-constrained graphs have a large tree-width: it is known that a graph with average degree at least Δ_{avg} and girth at least g has a tree width as $\Omega\left(\frac{1}{g+1}(\Delta_{\text{avg}}-1)^{\lfloor (g-1)/2\rfloor}\right)$ [11]. Thus, learning is non-trivial for graphical models Markov on girth-constrained graphs, even when there are no latent variables due to their large treewidth [27].

2.3. *Local Convergence to a Tree Limit.* This work establishes tractable learning when the graphical model converges locally to a tree limit. A sufficient condition for the existence of such limits is the regime of *correlation decay*[1], which refers to the property that there are no long-range correlations in the model [25, 33, 45]. This regime is also known as the *uniqueness regime* since under such an assumption, the marginal distribution at a node is asymptotically independent of the configuration of a growing boundary.

We tailor the definition of correlation decay to node neighborhoods and provide the definition below. Given a graph $G = (W, E)$ and a distribution $P_{\mathbf{X}_W|G}$ Markov on it, and any subset $A \subset W$, let $P_{\mathbf{X}_A|G}$ denote the marginal distribution of variables in A. For some subgraph $F \subset G$, let $P_{\mathbf{X}_A|F}$ denote the marginal distribution on A obtained by setting the potentials of edges in $G \setminus F$ to zero. Thus, $P_{\mathbf{X}_A|F}$ is Markov on graph F. Let $\mathcal{N}[i;G] := \mathcal{N}(i;G)\cup i$ denote the closed neighborhood of node i in G. For any two sets $A_1, A_2 \subset W$, let $\text{dist}(A_1,A_2) := \min_{i\in A_1, j\in A_2}\text{dist}(i,j)$ denote the minimum graph distance[2]. Let $B_l(i)$ denote the set of nodes within graph distance l from node i and $\partial B_l(i)$ denote the boundary nodes, i.e., exactly at distance l from node i. Let $F_l(i;G) := G(B_l(i))$ denote the induced subgraph on $B_l(i)$. For any distributions P, Q, let $\|P-Q\|_1$ denote the ℓ_1 norm.

[1]Technically, correlation decay can be defined in multiple ways [33, p. 520] and the notion we use is the uniqueness or the weak spatial mixing condition.

[2]We distinguish between the terms *graph distance* and *information distances*. The former refers to the number of edges on the shortest path connecting the two nodes on the (unweighted) graph, while the latter refers to the quantity in (8).

Definition 1 (Correlation Decay). *A distribution $P_{\mathbf{X}_{W_m}|G}$ Markov on graph $G_m = (W_m, E_m)$ is said to exhibit correlation decay with a non-increasing rate function $\zeta(\cdot) > 0$ if for all $l, m \in \mathbb{N}$,*

$$\|P_{\mathbf{X}_A|G_m} - P_{\mathbf{X}_A|F_l(i;G_m)}\|_1 \leq \zeta(\text{dist}(A, \partial B_l(i))), \quad \forall\, i \in W_m,\ A \subset B_l(i). \tag{3}$$

In words, the total variation distance[3] between the marginal distribution of a set A of a distribution Markov on G_m and the corresponding distribution Markov on subgraph $F_l(i; G_m)$ decays as a function of the graph distance to the boundary. This implies that for a class of functions $\zeta(\cdot)$, the effect of graph configuration beyond l hops from any node i has a decaying effect on the local marginal distributions.

For the class of Ising models in (2), the regime of correlation decay can be explicitly characterized, in terms of the maximum edge potential and the maximum degree of the graph, and this is studied in Section 4.2.

3. Background on Latent Tree Models. We first recap the results for latent tree models which will subsequently extended to more general latent graphical models. It is well known that tree-structured distributions Markov on a tree $T = (W, E)$ have a special form of factorization given by

$$P(\mathbf{x}_W) = \prod_{i \in W} P_{X_i}(x_i) \prod_{(i,j) \in T} \frac{P_{\mathbf{X}_{i,j}}(x_i, x_j)}{P_{X_i}(x_i) P_{X_j}(x_j)} \tag{4}$$

Comparing with general distributions, we note that tree distributions are directly parameterized in terms of pairwise marginal distributions on the edges. Similarly, a Markov distribution can be described on a rooted directed tree $\vec{T}$ with root $r \in W$, where the edges of $\vec{T}$ are directed away from the root. Let $\text{Pa}(i)$ denote the (unique) parent of node $i \neq r$ and $P_{X_i|X_{\text{Pa}(i)}}$ denote the corresponding conditional distribution. The Markov distribution is given by

$$P(\mathbf{x}_W) = P_{X_r}(x_r) \prod_{i \in W, i \neq r} P_{X_i|X_{\text{Pa}(i)}}(x_i | x_{\text{Pa}(i)}). \tag{5}$$

A Markov model is said to be *non-singular* [35, 44] if (a) For all $e \in \vec{T}$, the conditional distributions satisfy $0 < |\det(P_{X_i|X_{\text{Pa}(i)}})| < 1$ and (b) For all $i \in V$, $P_{X_i}(x) > 0$ for all $x \in \mathcal{X}$. A non-singular Markov model on an undirected tree T and its directed counterpart $\vec{T}$ are equivalent [35, 44]. Note that non-singularity is equivalent to positivity (i.e., bounded potential functions) for Markov tree models. In particular, Ising models on trees with bounded node and edge potentials are non-singular. This is because under positivity, there is positive probability for any global configuration of node states which implies that the conditional probability at a node given any of its neighbors cannot be degenerate.

Latent tree models or phylogenetic tree models are tree-structured graphical models in which a subset of nodes are hidden or latent. Our goal in this paper is to leverage on the techniques developed for learning latent tree models to analyze a more general class of latent graphical models.

[3] Recall that the total variation distance between two probability distributions P, Q on the same alphabet is given by $\frac{1}{2}\|P - Q\|_1$.

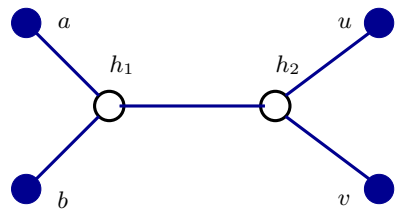


FIG 1. *Quartet $Q(ab|uv)$. See (7).*

3.1. *Learning Latent Tree Models.* Learning the structure of latent tree models is an extensively studied topic. A majority of structure learning methods (known as distance based methods) rely on the presence of an *additive tree metric*. The additive tree metric can be obtained by considering the pairwise marginal distributions of a tree structured joint distribution. For instance, the work in [34] considers the following metric for discrete distributions satisfying the non-singular condition

$$d(i,j) := -\log|\det(P_{\mathbf{X}_{i,j}})|, \quad \forall i,j \in V. \tag{6}$$

By non-singularity assumption, we have that $|\det(P_{\mathbf{X}_{i,j}})| > 0$ for all $i,j \in W^2$. The distance metric further simplifies for some special distributions, e.g. for symmetric Ising models, it is given by the negative logarithm of the correlation between the node pair under consideration [42].

3.1.1. *Quartet Based Methods.* A popular class of learning methods are based on the construction of *quartets* or *splits* (e.g., [9, 22, 34]), and various procedures to merge the inferred quartets. A quartet is a structure over four observed nodes, as shown in Fig.1. We now recap the classical quartet test operating on any additive tree metric. The path structure refers to the configuration of paths between the given nodes.

DEFINITION 2 (Quartet or Four-Point Condition on Trees). *Given an additive metric on a tree $[d(i,j)]_{i,j\in V}$, the tuple of four nodes $a,b,u,v \in V$ has the structure in Fig.1 iff.*

$$d(a,b) + d(u,v) < \min(d(a,u) + d(b,v), d(b,u) + d(a,v)), \tag{7}$$

and the structure in Fig.1 is denoted by $Q(ab|uv)$.

It is well known that the set of all quartets uniquely characterize a latent tree. In [22], it was shown that a subset of quartets, termed as *representative quartets*, suffices to uniquely characterize a latent tree. The set of representative quartets consists of one quartet for each edge in the latent tree with shortest (graph) distances between the observed nodes.

3.1.2. *Recursive Grouping.* We recap the recursive grouping $\mathsf{RG}(\widehat{\mathbf{d}}^n(V), \Lambda, \tau)$ method proposed in [16] (and its refinement in [1]). The method is based on a robust quartet test $\mathsf{Quartet}(\widehat{\mathbf{d}}^n, \Lambda)$ given in Algorithm 1. If the confidence bound is not met, a $\perp$ result is declared. In the first iteration of RG, the algorithm searches for node pairs which occur on the same side of all the quartets, output by the quartet test $\mathsf{Quartet}(\widehat{\mathbf{d}}^n, \Lambda)$ and declares them as siblings and introduces hidden variables. In later iterations of RG, sibling relationships between hidden variables are inferred through quartets involving their children. Finally, weak edges are merged and a tree (and more generally a forest) is output. We later use a modified version of recursive grouping method as a routine in our algorithm for estimating locally tree-like graphs. In the end, the neighboring nodes (at least one of which is hidden) are merged based on the threshold τ. See Section 4 for details.

Algorithm 1 $\mathsf{Quartet}(\widehat{\mathbf{d}}^n(V), \Lambda)$ test using distance estimates $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}(i,j)\}_{i,j\in V}$ and confidence bound Λ.

Input: Distance estimates between the observed nodes $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}(i,j)\}_{i,j\in V}$ and confidence bound Λ.Denote $(\cdot)_+ := \max(\cdot, 0)$.
Initialize set of quartets $\mathcal{Q}(V) \leftarrow \emptyset$.
for $\{i, j, i', j'\} \in V$ **do**
  if $(e^{-\widehat{d}(i,j)} - \Lambda)_+(e^{-\widehat{d}(i',j')} - \Lambda)_+ > (e^{-\widehat{d}(i,j')} + \Lambda)_+(e^{-\widehat{d}(i,j)} + \Lambda)_+$ **then**
    Declare Quartet: $\mathcal{Q}(V) \leftarrow Q(ij|i'j')$.
  end if
  if No quartet declared for $\{i, j, i', j'\}$ **then**
    $\perp_{i,j,i',j'}$ (Declare null).
  end if
end for

Algorithm 2 $\mathsf{RG}(\widehat{\mathbf{d}}^n(V), \Lambda, \tau)$ test using distance estimates $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}(i,j)\}_{i,j\in V}$, confidence bound Λ and threshold τ for merging nodes.

Input: Distance estimates between the observed nodes $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}(i,j)\}_{i,j\in V}$, confidence bound Λ and threshold τ. Let $\mathcal{C}(a)$ denote the children of node a.
Initialize $A \leftarrow V$, $\mathcal{C}(i) \leftarrow \{i\}$ for all $i \in V$ and $\mathcal{Q}(V) \leftarrow \mathsf{Quartet}(\widehat{\mathbf{d}}^n(A), \Lambda)$.
while $A \neq \emptyset$ **do**
  if $\exists\, i, j \in A$ s.t. for each $a \in \mathcal{C}(i)$ and $b \in \mathcal{C}(j)$, $c, d \notin \mathcal{C}(i) \cup \mathcal{C}(j)$, $\{ac|bd, ad|bc\} \notin \mathcal{Q}(V)$, i.e., a, b are on same side of all such quartets in $\mathcal{Q}(V)$. **then**
    Declare i, j as siblings and introduce hidden node h as parent and $\mathcal{C}(h) \leftarrow \mathcal{C}(i) \cup \mathcal{C}(j)$.
    Remove i, j from A and add h to A.
  else
    Sibling relationships cannot be further inferred. Break.
  end if
end while
Form forest $\widehat{T}$ based on sibling and child/parent relationships.
Compute distances between any two hidden nodes as average distance between their observed children.
Merge edges in $\widehat{T}$ of length less than τ and output $\widehat{T}$.

3.1.3. *Chow-Liu Grouping.* An alternative method, known as *Chow-Liu grouping* (CLGrouping), was proposed in [16]. Although the theoretical results for CLGrouping are similar to earlier results (e.g. [22]), experiments on both synthetic and real data sets revealed significant improvement over earlier methods in terms of likelihood scores and number of hidden variables added.

The CLGrouping method always maintains a candidate tree structure and progressively adds more hidden nodes in local neighborhoods. The initial tree structure is the *minimum spanning tree* (MST) over the observed nodes with respect to the tree metric. The method then considers neighborhood sets on the MST and constructs local subtrees (using quartet based method or any other tree reconstruction algorithm). This local reconstruction property of CLGrouping makes it especially attractive for reconstructing girth-constrained graphs.

4. Method and Guarantees for Structure Estimation.

4.1. *Overview of Algorithm.* We now describe our algorithm, which we term as LocalCLGrouping, for structure estimation of latent graphical models Markov on graphs with long cycles. The algorithm leverages on the Chow-Liu grouping algorithm developed for latent tree models [16], described in the previous section. The main intuition for learning a girth-constrained graph is based on reconstructing "local" parts of the graph which are acyclic and piecing them together. However, this

Algorithm 3 $\mathsf{LocalCLGrouping}(\widehat{\mathbf{d}}^n(V), \Lambda, \tau, r)$ for graph estimation using distance estimates $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}(i,j)\}_{i,j\in V}$, confidence bound Λ, threshold τ and distance parameter r.

Input: Distance estimates between the observed nodes $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}(i,j)\}_{i,j\in V}$, confidence bound Λ, threshold τ and bound r on distances used for local reconstruction. Let $B_r(v;\widehat{\mathbf{d}}^n) := \{u : \widehat{d}^n(u,v) \leq r\}$ and $\mathrm{MST}(A;\widehat{\mathbf{d}}^n)$ denotes the minimum spanning tree over $A \subset V$ based on edge weights $\widehat{\mathbf{d}}^n(A)$. Given a graph G, let $\mathrm{Leaf}(G)$ denote the set of nodes with unit degree. Let $\mathcal{N}[i;G]$ denote the closed neighborhood of node i in graph G. $\mathsf{RG}(\widehat{\mathbf{d}}^n(A), \Lambda, \tau)$ represents the recursive grouping method for building latent trees (see Section 3.1) over the set of nodes A using distance estimates $\widehat{\mathbf{d}}^n(A)$ with confidence bound Λ and threshold τ for merging nodes.

for $v \in V$ **do**
  $T_v \leftarrow \mathrm{MST}(B_r(v);\widehat{\mathbf{d}}^n)$.
end for
Initialize $\widehat{G}, \widehat{G}_0 \leftarrow \cup_v T_v$.
for $v \in V \setminus \mathrm{Leaf}(\widehat{G}_0)$ **do**
  $A \leftarrow \mathcal{N}[v;\widehat{G}]$.
  $S \leftarrow \mathsf{RG}(\widehat{\mathbf{d}}^n(A), \Lambda, \tau)$.
  $\widehat{G}(A) \leftarrow S$ (Replace subgraph over A with S in $\widehat{G}$)
end for
Output $\widehat{G}$.

approach has many challenges. First, it is not clear if the local acyclic pieces can be learnt efficiently since it requires the presence of an additive tree metric. This is addressed by considering models satisfying correlation decay (see Section 2.3). Second and a harder challenge involves merging the reconstructed local latent trees with provable guarantees due to the introduction of unlabeled latent nodes in different pieces. We circumvent this challenge by leveraging on the Chow-Liu grouping algorithm [16] and merging the different pieces before introducing the latent nodes.

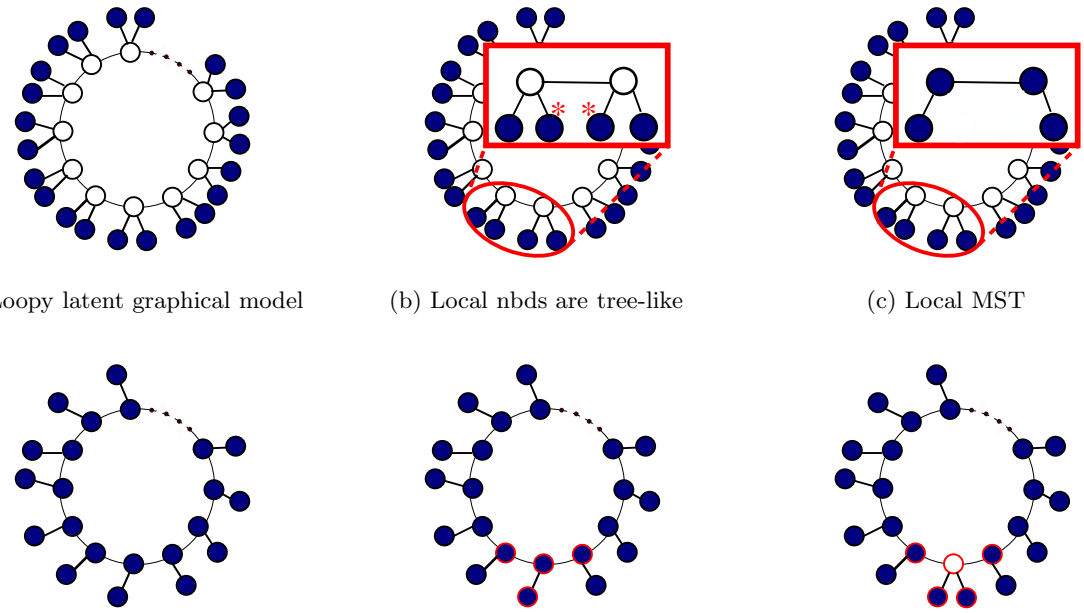
The algorithm is described in Algorithm 3. Let $\widehat{d}^n(i,j)$ denote the estimated distance between nodes i and j according to (6) using the empirical distribution $\widehat{P}^n_{\mathbf{X}_{i,j}}$ computed using n samples, i.e.,

$$\widehat{d}^n(i,j) := -\log|\det(\widehat{P}^n_{\mathbf{X}_{i,j}})|, \quad \forall i,j \in V. \tag{8}$$

The set of distance estimates $\widehat{\mathbf{d}}^n(V) := \{\widehat{d}^n(i,j) : i,j \in V\}$ are input to the algorithm along with a parameter r. Recall that $B_r(i;\widehat{\mathbf{d}}^n(V)) := \{j : \widehat{d}^n(i,j) \leq r\}$. For each observed node $i \in V$, the set of nodes $B_r(i;\widehat{\mathbf{d}}^n(V))$ is considered, and the minimum spanning tree is constructed. The graph estimate $\widehat{G}$ is initialized by taking the union of all the local minimum spanning trees. The latent nodes are now iteratively added by considering local neighborhoods of $\widehat{G}$ and using any latent tree algorithm for reconstruction (e.g. [16, 34]). Note that the running time is polynomial (in the number of nodes) as long as polynomial time algorithms are employed for local latent tree reconstruction.

The proposed method is efficient for practical implementation due to the "divide and conquer" feature, i.e., the local latent tree building operations can be parallelized to obtain speedups. For real datasets, a tradeoff between model complexity and fidelity is typically enforced by optimizing scores such as the Bayesian information criterion (BIC) [41]. Such criteria can be easily enforced through a greedy local search in each iteration of our method, and this limits the number of hidden variables added by our method. In our experiments in Section 6, we found that this method is fast to implement on real and synthetic datasets.

We subsequently establish the correctness of the proposed method under a set of natural conditions. We require that the parameter r, which determines the set $B_r(i;\mathbf{d})$ for each node i, needs to be chosen as a function of the depth δ (i.e., distance from a hidden node to its closest observed



(a) Loopy latent graphical model (b) Local nbds are tree-like (c) Local MST

(d) Merging of local MSTs (e) Local nbd on merged graph (f) Latent tree on local nbd

FIG 2. *Various steps of* LocalCLGrouping *method on a simple cycle, where observed variables are shaded. See Section 4.1.1.*

nodes) and girth g of the graph. In practice, the parameter r provides flexibility in tuning the length of cycles added to the graph estimate. When r is large enough, we obtain a latent tree, while for small r, the graph estimate can contain many short cycles (and potentially many components). In experiments, we evaluate the performance of our method for different values of r. The tuning of parameters Λ and τ has been previously discussed in the context of learning latent trees (e.g. [16, p. 1796]) and we leverage on those results here. For more details, see Section 6.

4.1.1. *Simple Example with a Single Cycle.* To demonstrate the steps of the above proposed method, consider the simple case of a single cycle of length g, where all the nodes on the cycle are hidden and each hidden node has two observed leaves, as shown in Fig.2a. When the cycle length g is sufficiently large, information distances on local neighborhoods are approximately additively, as depicted in Fig.2b. Moreover, in Fig.2b, let " * " denote the observed node closest to each hidden node (termed as its *surrogate*), in terms of information distance. The minimum spanning tree over the set of four nodes, which are zoomed in, corresponds to a chain shown in Fig.2c. Similarly, if in different local neighborhoods of observed nodes (based on a threshold on information distances), the surrogate relationships are similar (i.e., every hidden node has one of its children as its surrogate), then the local MSTs are simple chains, and their merging gives rise to graph G in Fig.2d. Now if a local neighborhood is selected on the merged graph G, as shown in Fig.2e, then we can discover the local latent tree structure based on information distances as shown in Fig.2f, since they are approximately additive. Similarly, when different neighborhoods on G are selected, local latent trees are discovered, and distances between nearby hidden nodes are computed. This way we recover the latent cycle graph in Fig.2a in the end.

4.2. *Results for Ising Models.* We first limit ourselves to providing asymptotic guarantees for the Ising model in (2), and then extend the results to non-asymptotic guarantees in general discrete distributions.

4.2.1. *Conditions for Recovery in Ising Models.* We present a set of natural conditions on the graph structure and model parameters under which our proposed method succeeds in structure estimation.

(A1) **Minimum Degree of Latent Nodes:** We require that all latent nodes have degree at least three.

(A2) **Distance Bounds:** Assume bounds on the edge potentials $\boldsymbol{\theta} := \{\theta_{i,j}\}$ of the Ising model:

$$\theta_{\min} \leq |\theta_{i,j}| \leq \theta_{\max}, \quad \forall\, (i,j) \in G. \tag{9}$$

Similarly assume bounded node potentials. We now define certain quantities which depend on the edge potential bounds. Given a distribution belonging to the class of Ising models P with edge potentials $\boldsymbol{\theta} = \{\theta_{i,j}\}$ and node potentials $\boldsymbol{\phi} = \{\phi_i\}$, consider its attractive counterpart $\bar{P}$ with edge potentials $\bar{\boldsymbol{\theta}} := \{|\theta_{i,j}|\}$ and node potentials $\bar{\boldsymbol{\phi}} := \{|\phi_i|\}$. Let $\phi'_{\max} := \max_{i \in V} \operatorname{atanh}(\bar{\mathbb{E}}(X_i))$, where $\bar{\mathbb{E}}$ is the expectation with respect to the distribution $\bar{P}$. Let $P(\mathbf{X}_{1,2}; \{\theta, \phi_1, \phi_2\})$ denote a distribution belonging to the class of Ising models on two nodes $\{1,2\}$ with edge potential θ and node potentials $\{\phi_1, \phi_2\}$. Our learning guarantees depend on $d_{\min}$ and $d_{\max}$ satisfying

$$d_{\min} \geq -\log |\det P(\mathbf{X}_{1,2}; \{\theta_{\max}, \phi'_{\max}, \phi'_{\max}\})|, \tag{10}$$

$$d_{\max} \leq -\log |\det P(\mathbf{X}_{1,2}; \{\theta_{\min}, 0, 0\})|, \tag{11}$$

$$\eta := \frac{d_{\max}}{d_{\min}}. \tag{12}$$

(A3) **Correlation Decay:** We assume correlation decay in the Ising model and require that

$$\alpha := \Delta_{\max} \tanh \theta_{\max} < 1, \quad \frac{\alpha^{g/2}}{\theta_{\min}^{\eta(\eta+1)+2}} = o(1), \tag{13}$$

where $\Delta_{\max}$ is the maximum node degree, g is the girth, $\theta_{\min}, \theta_{\max}$ are the minimum and maximum (absolute) edge potentials in the model and $o(1)$ is with respect to m, the number of nodes in the graph[4].

(A4) **Girth vs. Depth:** The depth δ characterizes how close the latent nodes are to observed nodes on graph G: for each hidden node $h \in H$, find a set of four observed nodes which form the shortest *quartet* with h as one of the middle nodes, and consider the largest graph distance in that quartet. The depth δ is the worst-case distance over all hidden nodes. We require the following tradeoff between the girth g and the depth δ:

$$\frac{g}{4} - \delta\eta\,(\eta+1) = \omega(1), \tag{14}$$

[4]Unless otherwise noted, the notations $O(\cdot), o(\cdot), \Omega(\cdot), \omega(\cdot)$ are with respect to m, the number of nodes in the graph.

Further, the parameter r in our algorithm is chosen as

$$r > \delta\,(\eta+1)\,d_{\max} + \epsilon, \quad \text{for some } \epsilon > 0, \quad \frac{g}{4} d_{\min} - r = \omega(1). \tag{15}$$

$(A1)$ is a natural assumption on the minimum degree of the hidden nodes for identifiability and has been imposed before for latent tree models [16]. Note that the latent nodes of degree two or lower can be marginalized to obtain an equivalent representation of the observed statistics.

$(A2)$ relates certain distance bounds to bounds on edge potentials. Intuitively, $d_{\min}$ and $d_{\max}$ are bounds on information distances given by the local tree approximation of the loopy model, and its precise definition is given in (18). Note that $e^{-d_{\max}} = \Omega(\theta_{\min})$ and $e^{-d_{\min}} = O(\theta_{\max})$.

$(A3)$ uses bounds on the edge potentials to impose correlation decay on the model. It is natural that the sample requirement of any graph estimation algorithm depends on the "weakest" edge characterized by the minimum edge potential $\theta_{\min}$. Further, the maximum edge potential $\theta_{\max}$ characterizes the presence/absence of long range correlations in the model. Moreover, $(A3)$ prescribes that the extent of correlation decay be strong enough (i.e., a small α and a large enough girth g) compared to the weakest edge in the model.

Conditions similar to $(A3)$ have been imposed before for graphical model selection in the regime of correlation decay when there are no hidden variables [3]. For instance, in [3], an upper bound is imposed on the edge potentials to limit the effect of long paths on local conditional independence tests. A lower bound on edge potentials is needed for edges to pass the conditional independence test.

$(A4)$ provides the tradeoff between the girth g and the depth δ. Intuitively, the depth needs to be smaller than the girth to avoid encountering cycles during the process of graph reconstruction. Recall that the parameter r in our algorithm determines the neighborhood over which local MSTs are built in the first step. It is chosen such that it is roughly larger than the depth δ in order for all the hidden nodes to be discovered. The upper bound on r ensures that the distortion from an additive metric is not too large. The parameters for latent tree learning routines (such as confidence bounds for quartet tests) are chosen appropriately depending on $d_{\min}$ and $d_{\max}$. See Section 4.3.

4.2.2. *Guarantees for Ising Models.* We now establish that the proposed method correctly estimates the graph structure of an Ising model in high dimensions. Recall that δ is the depth (distance from a hidden node to its closest observed nodes), $\theta_{\min}$ is the minimum (absolute) edge potential and $\eta = \frac{d_{\max}}{d_{\min}}$ is the ratio of distance bounds.

THEOREM 1 (Structural Consistency for Ising Models). *Under* $(A1)$–$(A4)$, *the probability that* LocalCLGrouping *is structurally consistent tends to one, when the number of samples scales as*

$$n = \Omega\left(\theta_{\min}^{-\delta\eta(\eta+1)-2} \log p\right). \tag{16}$$

Proof: See the supplementary material. □

Remarks.

1. For learning Ising models on locally tree-like graphs, the sample complexity is dependent both on the minimum edge potential $\theta_{\min}$ and on the depth δ. Our method is efficient in high dimensions since the sample requirement is only logarithmic in the number of nodes p.
2. **Dependence on Maximum Degree:** For the correlation decay to hold ($A3$), we require $\theta_{\min} \leq \theta_{\max} = \Theta(1/\Delta_{\max})$. This implies that the sample complexity is at least $n = \Omega(\Delta_{\max}^{\delta\eta(\eta+1)+2} \log p)$.
3. **Comparison with Fully Observed Models:** In the special case when all the nodes are observed ($\delta = 1$) and the graph is locally tree-like, we strengthen the results for our method and establish that the sample complexity for graph estimation is $n = \Omega(\theta_{\min}^{-2} \log p)$. This matches the best known sample complexity for learning fully observed Ising models [3, 26]. The sample complexity result holds for a modified version of LocalCLGrouping: threshold r is applied to the information distances at each node and local MSTs are formed as before. The threshold r can be chosen as $r = d_{\max} + \epsilon$, for some $\epsilon > 0$. The graph estimate is obtained as the union of local MSTs and local latent tree routines are not implemented in this case. We prove an improved sample complexity in this special case which matches the best known bounds.
4. **Comparison with Learning Latent Trees:** Our method is an extension of latent tree methods for learning locally tree-like graphs. The sample complexity of our method matches the sample requirements for learning general latent tree models [16, 22, 34]. Thus, we establish that learning locally tree-like graphs is akin to learning latent trees in the regime of correlation decay.

4.3. *Extension to General Discrete Models.* We now extend the results to general discrete models and provide non-asymptotic sample requirement guarantees for success of our proposed method.

Local Tree Approximation. We first define the notion of a local tree metric $\mathbf{d}_{\text{tree}}(V)$ computed by limiting the model to acyclic neighborhood subgraphs between the respective node pairs. Given a graph $G = (W, E)$, let $\text{tree}(i, j; G) := G(B_l(i) \cup B_l(j))$, for $l = \lfloor g/2 \rfloor - 1$, denote the induced subgraph on $B_l(i) \cup B_l(j)$, where g is the girth of the graph. Recall that $B_l(i; G)$ denotes the set of nodes within graph distance l from i in G. When $l < g/2 - 1$ no cycles are encountered and thus the induced subgraph $\text{tree}(i, j; G)$ is acyclic. Recall that $P_{\mathbf{X}_{i,j}|G}$ denotes the pairwise marginal distribution between i and j induced by the distribution $P(\mathbf{x}_W)$ Markov on graph G. Let $P_{\mathbf{X}_{i,j}|\,\text{tree}(i,j)}$ denote the pairwise marginal distribution between i and j induced by considering only the subgraph $\text{tree}(i, j; G) \subset G$. Denote

$$d(i, j; \text{tree}) := -\log |\det P_{\mathbf{X}_{i,j}|\,\text{tree}(i,j)}|, d(i, j; G) := -\log |\det P_{\mathbf{X}_{i,j}|G}|. \tag{17}$$

Denote $\mathbf{d}_{\text{tree}}(V) := \{d(i, j; \text{tree}) : i, j \in V\}$ and $\mathbf{d}(V) := \{d(i, j; G) : i, j \in V\}$. Note that for loopy graphs in general, $d(i, j; G)$ is different from $d(i, j; \text{tree})$. The learner has access only to the empirical versions $\widehat{\mathbf{d}}(V)$ of the distances $\mathbf{d}(V)$, and thus the learner cannot estimate $\mathbf{d}_{\text{tree}}(V)$. However, we use $\mathbf{d}_{\text{tree}}(V)$ to characterize the performance of our algorithm, we list the relevant assumptions below.

4.3.1. *Conditions on the Model Parameters.*

($B1$) **Minimum Degree:** The minimum degree of any hidden node in the graph is three.

($B2$) **Bounds on Local Tree Metric:** Given a distribution $P_{\mathbf{X}_W|G}$ Markov on graph G, the pairwise marginal distribution $P_{\mathbf{X}_{i,j}|\,\mathrm{tree}(i,j)}$ between any two neighbors $(i,j) \in G$ are non-singular and the distances $d(i,j;\mathrm{tree}) := -\log|\det P_{\mathbf{X}_{i,j}|\,\mathrm{tree}(i,j)}|$ satisfy

$$0 < d_{\min} \leq d(i,j;\mathrm{tree}) \leq d_{\max} < \infty, \quad \forall (i,j) \in G, \quad \eta := \frac{d_{\max}}{d_{\min}}, \tag{18}$$

for suitable parameters $d_{\min}$ and $d_{\max}$.

($B3$) **Regime of Correlation Decay:** The pairwise statistics of the distribution converge locally to a tree limit according to Definition 1 with function $\zeta(\cdot)$ in (3) satisfying

$$0 \leq \zeta\left(\frac{g}{2} - \frac{r}{d_{\min}} - 1\right) < \frac{\upsilon}{|\mathcal{X}|^2}, \tag{19}$$

where g is the girth, r is the distance bound parameter in $\mathsf{LocalCLGrouping}$, $|\mathcal{X}|$ is the dimension of each variable, $d_{\min}, d_{\max}$ are the distance bounds in (18) and

$$\upsilon := \min\left(d_{\min}, 0.5e^{-r}(e^{d_{\min}} - 1), e^{-0.5d_{\max}(\frac{r}{d_{\min}}+2)}, \frac{g}{4}d_{\min} - r, r - d_{\max}\delta(\eta+1)\right). \tag{20}$$

($B4$) **Confidence Bound for Quartet Test:** The confidence bound in $\mathsf{Quartet}(\widehat{\mathbf{d}}, \Lambda)$ routine in Algorithm 1 is chosen as

$$\Lambda = \exp\left[-\frac{d_{\max}}{2}(\frac{r}{d_{\min}} + 2)\right]. \tag{21}$$

($B5$) **Threshold for Merging Nodes:** The threshold τ in $\mathsf{RG}(\widehat{\mathbf{d}}, \Lambda, \tau)$ routine in Algorithm 2 is chosen as

$$\tau = \frac{d_{\min}}{2} - |\mathcal{X}|^2\zeta(\frac{g}{2} - 1) > 0, \tag{22}$$

where $|\mathcal{X}|$ is the dimension of the variable at each node and $\zeta(\cdot)$ is the correlation decay function according to (3).

($B1$) is a natural assumption on the minimum degree of the hidden nodes for identifiability, which is also needed for latent trees. The assumption ($B2$) states that every edge has bounded distances under local tree approximations. Recall that in the special case of Ising models, this can be expressed via bounds on edge potentials. The assumption ($B3$) on correlation decay imposes a constraint on the rate function $\zeta(\cdot)$, in terms of the girth of the graph g, the distance threshold r used by the proposed method, the distance bounds $d_{\min}$ and $d_{\max}$ and depth δ. ($B3$) implies that we require that the depth δ satisfies

$$\frac{g}{4}d_{\min} > \delta\,(\eta+1)\,d_{\max}. \tag{23}$$

Similarly, ($B3$) imposes constraints on the parameter r used by the proposed algorithm for building local minimum spanning trees in the first step. ($B3$) implies that r needs to be chosen as

$$\delta\,(\eta+1)\,d_{\max} < r < \frac{g}{4}d_{\min} - r. \tag{24}$$

Intuitively, the above constraint implies that r is relatively small compared to the girth of the graph and large enough for every hidden node to be discovered. This enables the proposed algorithm to correct reconstruct latent trees locally.

The confidence bound constraint in $(B4)$ is based on the concentration bounds for the empirical distances. The threshold for merging nodes in $(B5)$ ensures that spurious hidden nodes are not added. These conditions are inherited from latent tree algorithms.

4.4. *Guarantees for the Proposed Method.* We now establish that the LocalCLGrouping algorithm is structurally consistent under the above conditions.

THEOREM 2 (Structural Consistency of LocalCLGrouping). *Under assumptions* $(B1)$-$(B5)$*, the* LocalCLGrouping *algorithm is structurally consistent with probability at least* $1-\kappa$*, for any* $\kappa > 0$*, when the sample size* n *satisfies*

$$n > \frac{2|\mathcal{X}|^2}{(\upsilon - |\mathcal{X}|^2 \zeta(\frac{g}{2} - \frac{r}{d_{\min}} - 1))^2} \left(4 \log p + |\mathcal{X}| \log 2 - \log \frac{\kappa}{7}\right), \tag{25}$$

where υ *is given by* (20).

Remarks:

1. We provide PAC guarantees for reconstructing latent graphical models on girth-constrained graphs. The conditions for success imposed on the girth of the graph are relatively mild. We require that the girth be roughly larger than the depth and that the correlation decay function $\zeta(\cdot)$ be sufficiently strong $(B3)$. Thus, learning girth-constrained graphs is akin to learning latent tree models (in terms of sample and computational complexities) under a wide range of conditions.
2. One notable additional condition required for learning girth-constrained graphs in contrast to latent trees is the requirement of correlation decay $(B3)$. However, we note that this is only a sufficient condition, and not necessary for learnability. For instance, the result in [19] establishes that the pairwise statistics converges locally to a tree limit for all attractive Ising models with strictly positive node potentials, but without any additional constraints on the parameters. Our results and analysis hold in such scenarios since we only require local convergence to a tree metric.
3. The results above are applicable for discrete models but can be extended to Gaussian models using the notion of *walk-summability* in place of correlation decay according to (3) (see [2]) and the negative logarithm of the correlation coefficient as the distance metric (see [16]). The results can also be extended to more general linear models such as multivariate Gaussian model, Gaussian mixtures and so on, along the lines of [1].

Proof: The detailed proof is given in the supplementary material. It consists of the following main steps:

1. We first prove correctness of LocalCLGrouping under the tree limit (i.e., distances $\mathbf{d}_{\text{tree}}(V) := \{d(i,j;\text{tree})\}_{i,j\in V}$) and then show sample-based consistency. The latter is based on concentration bounds, along the lines of analysis for latent tree models [22, 34], with an additional distortion introduced due to the presence of a loopy graph.

2. We now briefly describe the proof establishing the correctness of LocalCLGrouping algorithm under $\mathbf{d}_{\text{tree}}$ in girth-constrained graphs. Intuitively, the distances $d(i,j;\text{tree})$ correspond to a tree metric when the graph distance $\text{dist}(i,j) < g/2 - 1$, where g is the girth. Since LocalCLGrouping infers latent trees only locally, it avoids running into cycles and thus correctly reconstructs the local latent trees. The initialization step in LocalCLGrouping corresponds to the correct merge of this local latent trees under the assumptions on parameter r in (24) and the correctness of LocalCLGrouping is established.

□

4.4.1. *Guarantees under Uniform Sampling.* We have so far given guarantees for graph reconstruction, given an arbitrary set of observed nodes in the graph. We now specialize the results to the case when there is a uniform sampling of nodes and provide learning guarantees. This analysis provides intuitions on the relationship between the fraction of sampled nodes and the resulting learning performance.

Consider an ensemble of graphs on m nodes with girth at least g and minimum degree $\Delta_{\min} \geq 3$ and maximum degree $\Delta_{\max}$. Let $\rho := \frac{p}{m}$ denote the uniform sampling probability for selecting observed nodes. We have the following result on the depth δ. Define a constant $\epsilon_0 > 0$ as

$$\epsilon_0 = -\frac{\log(4m\Delta_{\max}(1-\rho)^{(\Delta_{\min}-1)^{g/2}})}{\log m}. \tag{26}$$

Lemma 1 (Depth Under Uniform Sampling). *Given uniform sampling probability of ρ, for any $\epsilon \leq \max(0, \epsilon_0)$,*

$$\delta < \tfrac{1}{\log(\Delta_{\min}-1)} \left(\log \left[\frac{\log(4m^{1+\epsilon}\Delta_{\max})}{|\log(1-\rho)|} \right] \right) \quad w.p. \geq 1 - m^{-\epsilon}. \tag{27}$$

Proof: The proof is by straightforward arguments on binomial random variables and the union bound. See the supplementary material. □

Remarks:

1. Assuming that the girth satisfies $g > 2\delta(1 + d_{\max}/d_{\min})$ w.h.p., when the sampling probability and the degrees are both constant, then

$$\rho = \Theta(1),\ \Delta_{\min}, \Delta_{\max} = O(1) \Rightarrow \delta = O(\log\log m) \Rightarrow n = \Omega(\text{poly}(\log m)),\ \text{w.h.p.},$$

where $\text{poly}(\log m)$ refers to a polylogarithmic dependence in m. On the other hand, with vanishing sampling probability, for $\beta \in [0,1)$, we have

$$\rho = \Theta(m^{\beta-1}),\ \Delta_{\min}, \Delta_{\max} = O(1) \Rightarrow \delta = O(\log m) \Rightarrow n = \Omega(\text{poly}(m)),\ \text{w.h.p.}$$

2. Recall that for Ising models, the best-case sample complexity of LocalCLGrouping for structural consistency (when $\eta = 1$ and $\theta_{\min} = \theta_{\max} = \Theta(1/\Delta_{\max})$) scales as

$$n = \Omega(\Delta_{\max}^{2(\delta+1)} \log p).$$

Thus, under uniform sampling, the sample complexity required for consistency scales as

$$n = \Omega\left(\Delta_{\max}^2 \left(\frac{\log p}{|\log(1-\rho)|}\right)^{4\frac{\log \Delta_{\max}}{\log(\Delta_{\min}-1)}} \log p\right).$$

For the special case when the graph is regular ($\Delta_{\min} = \Delta_{\max}$), this reduces to

$$n = \Omega\left(\Delta_{\max}^2 \rho^{-2} (\log p)^3\right). \tag{28}$$

5. Necessary Conditions for Graph Estimation. We have so far provided sufficient conditions for recovering latent graphical models Markov on girth-constrained graphs. We now provide necessary conditions on the number of samples required by any algorithm to reconstruct the graph. Let $\widehat{G}_n : (\mathcal{X}^{|V|})^n \to \mathcal{G}_m$ denote any deterministic graph estimator using n i.i.d. samples from node set V and $\mathcal{G}_m$ is the set of all possible graphs on m nodes.

We first define the notion of the graph edit distance based on inexact graph matching [10]. Let $G, \widehat{G}$ be two graphs with common labeled node set V and unlabeled node sets U and $\widehat{U}$. Without loss of generality, let $|U| \geq |\widehat{U}|$ and add $|U| - |\widehat{U}|$ number of isolated nodes to $\widehat{G}$. Let $\mathbf{A}_G, \mathbf{A}_{\widehat{G}}$ be the resulting adjacency matrices. Then the edit distance between $G, \widehat{G}$ is defined as

$$\text{dist}(\widehat{G}, G; V) := \min_{\pi} ||\mathbf{A}_{\widehat{G}} - \pi(\mathbf{A}_G)||_1,$$

where π is any permutation on the unlabeled nodes while keeping the labeled node set V fixed.

In other words, the edit distance is the minimum number of entries that are different in $\mathbf{A}_{\widehat{G}}$ and in any permutation of $\mathbf{A}_G$ over the unlabeled nodes. In our context, the labeled nodes correspond to the observed nodes V while the unlabeled nodes correspond to latent nodes H. We now provide necessary conditions for graph reconstruction up to certain edit distance.

THEOREM 3 (Necessary Condition). *For any deterministic estimator $\widehat{G}_m : (\mathcal{X}^{m^\beta})^n \mapsto \mathcal{G}_m$ based on n i.i.d. samples from m^β observed nodes $\beta \in [0,1]$ of a latent graphical model Markov on graph G_m on m nodes with girth g, minimum degree $\Delta_{\min}$ and maximum degree $\Delta_{\max}$, for all $\epsilon > 0$, we have*

$$\mathbb{P}[\text{dist}(\widehat{G}_m, G_m; V) > \epsilon m] \geq 1 - \frac{|\mathcal{X}|^{nm^\beta} m^{(2\epsilon+1)m} 3^{\epsilon m}}{m^{0.5\Delta_{\min} m}(m - g\Delta_{\max}^g)^{0.5\Delta_{\min} m}}, \tag{29}$$

under any sampling process used to choose the observed nodes.

Proof: The proof is based on counting arguments. See the supplementary material for details. □

Remarks:

1. The above result states that roughly

$$n = \Omega(\Delta_{\min} m^{1-\beta} \log m) = \Omega\left(\frac{\Delta_{\min}}{\rho} \log p\right) \tag{30}$$

samples are required for structural consistency. Thus, when $\beta = 1$ (constant fraction of observed nodes), logarithmic number of samples are necessary while when $\beta < 1$ (vanishing fraction of observed nodes), polynomial number of samples are necessary for reconstruction. From (28), recall that for Ising models, under uniform sampling of observed nodes, the best-case sample complexity of LocalCLGrouping (for homogeneous models on regular graphs with degree Δ and $\theta_{\min} = \theta_{\max} = \Theta(1/\Delta)$) scales as

$$n = \Omega(\Delta^2 \rho^{-2} (\log p)^3),$$

and thus, nearly matches the lower bound on sample complexity in (30).

6. Experiments. In this section we present experimental results on real and synthetic data. We evaluate performance in terms of perplexity, predictive perplexity and topic coherence, used frequently in topic modeling. In addition, we also study tradeoff between model complexity and data fitting through the Bayesian information criterion (BIC) [41]. Experiments are conducted using the 20 newsgroup data set, monthly stock returns from the S&P 100 companies and synthetic data. The datasets, software code and results are available at http://newport.eecs.uci.edu/anandkumar.

6.1. *Experimental Setup.*

Synthetic data. We generate samples from an Ising model Markov on a cycle (see Fig.2) with a fixed depth $\delta = 1$, a fixed latent node degree $\Delta = 4$, and different girths $g = 10, 20, 30, .., 100$. The node potentials are kept at zero, while the edge potentials are chosen randomly in the range $[0.05, 0.2]$. This ensures that the model remains in the regime of correlation decay since the critical potential $\theta^* = \text{atanh}(\Delta^{-1}) = 0.2554 > 0.2$.

Newsgroup data. We employ latent graphical models for topic modeling, i.e., modeling the relationships between various words co-occurring in documents. Each hidden variable in the model can be thought of as representing a topic, and topics and words in a document are drawn jointly from the graphical model. For a latent tree graphical model, topics and words are constrained to form a tree, while loopy models relax this assumption. We consider $n = 16,242$ binary samples of $p = 100$ keywords selected from the 20 newsgroup data. Each binary sample indicates the appearance of the given words in each posting. These samples are divided in to two equal groups, training and test sets for learning and testing purposes.

S&P data. We also employ latent graphical models for financial modeling and in particular, for estimating the dependencies between the stock trends of different companies. The data set consists of monthly stock returns of $p = 84$ companies[5] listed in S&P 100 index from 1990 to 2007. Experiments with this dataset allows us to demonstrate the performance of our algorithm on data using a Gaussian graphical model. The Gaussian model is a simplifying

[5]The 16 companies added after 1990 are dropped from the list of 100 companies listed in S&P 100 stock index for this analysis.

assumption but reveals interesting relationships between the companies. We note that more sophisticated kernel models can indeed be used in place of the Gaussian approximation, e.g. [43].

Methods. We consider a regularized variant of the method proposed earlier for latent graphical model selection. Here, in every iteration, the decision to add hidden variables to a local neighborhood is based on the improvement of the overall BIC score. In other words, when a local latent tree is constructed in each iteration of Algorithm 3, the overall BIC score is evaluated and the latent tree is accepted only if the score improves; otherwise it is rejected and the previous configuration is retained. Similar regularization has been previously employed for learning latent trees in [16]. This allows us to tradeoff model complexity and data fitting. In addition, we obtain better generalization by avoiding overfitting. Note that our proposed method only deals with structure estimation and we use expectation maximization (EM) for parameter estimation. For the newsgroup data we compare the proposed method with the LDA model[6].

Implementation. The above method is implemented in MATLAB. We used the modules for LBP, made available with UGM[7] package. The LDA models are learnt using the lda package[8].

Threshold selection r for our method. Recall that the parameter r in our method controls the size of neighborhoods over which the local MSTs are constructed in the first step of our method. We earlier presented ranges of r, where recovery of the loopy structure is theoretically guaranteed (w.h.p). However, in practice, this range is unknown, since the model parameters are unknown to the learner, and also since there is no ground truth with respect to real datasets. Here, we present intuitive criterion for selecting the threshold based on the BIC score. We choose the range for threshold r as

$$r_{\max} := \max_{(i,j)\in V\times V} d(i,j)\,,\; r_{\min} := \max_{j\in V}\min_{i\in V} d(i,j), \tag{31}$$

thereby disallowing disconnected components in the output graph. Note that if we choose $r \geq r_{\max}$, then the output is a latent tree. In our experiments, we choose one value above $r_{\max}$ to find a reference tree model and compare it with other outcomes. For the 20 newsgroup dataset, we find that $r_{\min} = 2.3678$ and $r_{\max} = 12.2692$. Therefore, we choose $r \in \{3, 5, 7, 9, 11, 13\}$ for our experiments on newsgroup data. For the monthly stock returns data, $r_{\min} = 1.0337$ and $r_{\max} = 8.1172$, and we choose r from 1.1 to 8.2. The tuning of parameters Λ and τ has been previously discussed in the context of learning latent trees (e.g. [16, p. 1796]) and we leverage on those results here.

Performance Evaluation. We evaluate performance based on the test perplexity [37] given by

$$\text{Perp-LL} := \exp\left[-\frac{1}{np}\sum_{k=1}^{n}\log P(\mathbf{x}^{\text{test}}(k))\right], \tag{32}$$

[6]Typically, LDA models the counts of different words in documents. Here, since we have binary data, we consider a binary LDA model where the observed variables are binary.

[7]These codes can be downloaded from http://www.di.ens.fr/~mschmidt/Software/UGM.html

[8]http://chasen.org/~daiti-m/dist/lda/

where n is the number of test samples and p is the number of observed variables (i.e., words). Thus the perplexity is monotonically decreasing in the test likelihood and a lower perplexity indicates a better generalization performance. Along the lines of (32), we also evaluate the predictive perplexity [6]

$$\text{Pred-Perp-LL} := \exp\left[-\frac{1}{np}\sum_{k=1}^{n}\log P(\mathbf{x}_{\text{pred}}^{\text{test}}(k)|\mathbf{x}_{\text{obs}}^{\text{test}}(k))\right], \tag{33}$$

where a subset of word occurrences $\mathbf{x}_{\text{obs}}^{\text{test}}$ is observed in test data and the performance of predicting the rest of words is evaluated. In our experiments, we randomly select half the words in test samples.

We also consider regularized versions of perplexity that capture tradeoff between model complexity and likelihood, given by

$$\text{Perp-BIC} := \exp\left[-\frac{1}{np}\,\text{BIC}(\mathbf{x}^{\text{test}})\right], \tag{34}$$

where the BIC score [41] is defined as

$$\text{BIC}(\mathbf{x}^{\text{test}}) := \sum_{k=1}^{n}\log P(\mathbf{x}^{\text{test}}(k)) - 0.5(\text{df})\log n, \tag{35}$$

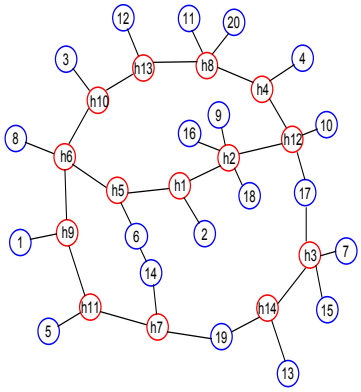
where df is the degrees of freedom in the model. For a graphical model, we set $\text{df}^{\text{GM}} := m + |E|$, where m is the total number of variables (both observed and hidden) and $|E|$ is the number of edges in the model. For the LDA model, we set $\text{df}^{\text{LDA}} := (p(m-p)-1)$, where p is the number of observed variables (i.e., words) and $m-p$ is the number of hidden variables (i.e., topics). This is because a LDA model is parameterized by a $p \times (m-p)$ topic probability matrix and a $(m-p)$-length Dirichlet prior. Thus, the BIC perplexity in (34) is monotonically decreasing in the BIC score, and a lower BIC perplexity indicates better tradeoff between model complexity and data fitting. However, the likelihood and BIC score in (32) and (34) are not tractable for exact evaluation in general graphical models since they involve the partition function. We employ loopy belief propagation (LBP) to evaluate them[9]. Note that it is exact on a tree model and approximate for loopy models. Along the lines of predictive perplexity in (33), we also consider its regularized version

$$\text{Pred-Perp-BIC} := \exp\left[-\frac{1}{np}\,\text{BIC}(\mathbf{x}_{\text{pred}}^{\text{test}}|\mathbf{x}_{\text{obs}}^{\text{test}})\right], \tag{36}$$

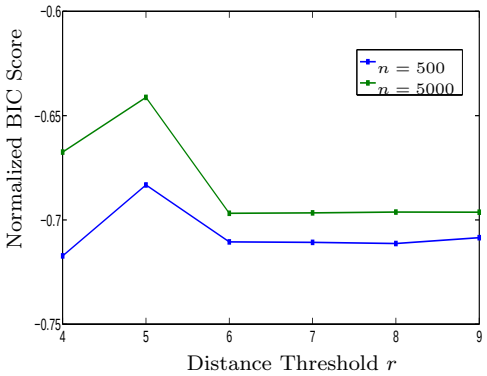
where the conditional BIC score is given by

$$\text{BIC}(\mathbf{x}_{\text{pred}}^{\text{test}}|\mathbf{x}_{\text{obs}}^{\text{test}}) := \sum_{k=1}^{n}\log P(\mathbf{x}_{\text{pred}}^{\text{test}}(k)|\mathbf{x}_{\text{obs}}^{\text{test}}(k)) - 0.5(\text{df})\log n, \tag{37}$$

[9]The likelihood is evaluated using $P(\mathbf{x}_V) = \frac{P(\mathbf{x}_{V\cup H})}{P(\mathbf{x}_H|\mathbf{x}_V)}$, where $P(\mathbf{x}_H|\mathbf{x}_V)$ and $P(\mathbf{x}_{V\cup H})$ are computed using LBP, which is exact for trees. The above expression holds for any configuration of hidden variables $\mathbf{x}_H$, however we use the most likely hidden state to avoid numerical issues.



(a) Output under threshold $r = 8$



(b) Normalized BIC vs. threshold r for different samples sizes

FIG 3. *Results for synthetic data with girth $g = 10$ using the proposed method.*

In addition, we also evaluate topic coherence, frequently considered in topic modeling. It is based on the average pointwise mutual information (PMI) score

$$\overline{\text{PMI}} := \frac{1}{45|H|} \sum_{h \in H} \sum_{\substack{i,j \in \mathcal{A}(h) \\ i<j}} \text{PMI}(X_i; X_j), \quad \text{PMI}(X_i; X_j) := \log \frac{P(X_i = 1, X_j = 1)}{P(X_i = 1)P(X_j = 1)}, \tag{38}$$

where the set $\mathcal{A}(h)$ represents the "top-10" words associated with topic $h \in H$. The number of such word pairs for each topic is $\binom{10}{2} = 45$, and is used for normalization. In [38], it is found that the PMI scores are a good measure of human evaluated topic coherence when it is computed using an external corpus. It is also observed that using a related external corpus gives a high PMI. Hence, in our experiments, we choose a corpus containing news articles from the NYT articles bag-of-words dataset. This dataset has a vocabulary of 102660 words from 300,000 separate articles [23]. For LDA models, the top 10 words for each topic are selected based on the topic probability vector. For latent graphical models, we use the criterion of information distances on the learnt model to select the 10 nearest words for each topic.

6.2. *Experimental Results.*

Results for Synthetic Data. We observe that our method outputs graphs with similar number of latent variables as the ground truth when r is chosen close to the bound $r_{\max}$, defined in (31). On the other hand, lower values of r leads to more cycles and hidden variables in the output graph. The normalized BIC scores (normalized with respect to n and p) of the loopy graphs improve with the number of samples n, as shown in Figure 3b. This is expected since the data becomes less noisy with more samples. Figure 3b shows an overall improvement in the normalized BIC score with increasing number of samples n for different thresholds r. Figure 3b shows the variation of normalized BIC scores for graphs learnt using thresholds $r = 4$ to 9 with girth $g = 10$. We observe that the normalized BIC score decreases for the lowest threshold ($r = 4$), where the output graph shows a significant increase in latent nodes and edges, resulting in overfitting, and higher thresholds have better BIC. However, once the threshold results in a tree model, the BIC degrades since the cycles are no longer present.

Method	r	Hidden	Edges	PMI	Perp-LL	Perp-BIC	Pred-Perp-LL	Pred-Perp-BIC
Proposed	3	55	265	0.2638	1.1533	1.1560	1.0695	1.0720
Proposed	5	39	293	0.4875	1.1567	1.1594	1.0424	1.0448
Proposed	7	32	183	0.4313	1.1498	1.1518	1.0664	1.0682
Proposed	9	24	129	0.6037	1.1543	1.1560	1.0780	1.0795
Proposed	11	26	125	0.4585	1.1555	1.1571	1.0787	1.0802
Proposed	13	24	123	0.4289	1.1560	1.1576	1.0788	1.0803
LDA	NA	10	NA	0.2921	1.1480	1.1544	1.1623	1.1656
LDA	NA	20	NA	0.1919	1.1348	1.1474	1.1572	1.1638
LDA	NA	30	NA	0.1653	1.1421	1.1612	1.1616	1.1715
LDA	NA	40	NA	0.1470	1.1494	1.1752	1.1634	1.1767

TABLE 1

Comparison of proposed method under different thresholds (r) with LDA under different number of topics (i.e., number of hidden variables) on 20 newsgroup data. For definition of perplexity and predictive perplexity based on test likelihood and BIC scores, and PMI, see (32), (33), (34), (36) and (38).

Topic 16	Topic 18	Topic 12	Topic 1	Topic 8
lunar	disk	card	god	software
moon	drive	video	jesus	pc
orbit	dos	windows	bible	computer
solar	memory	driver	christian	system
mission	windows	graphics	religion	dos
satellite	pc	dos	earth	windows
earth	software	version	question	disk
shuttle	scsi	ftp	fact	science
mars	computer	pc	jews	drive
space	system	disk	evidence	university

TABLE 2

Top 10 topic words from selected topics in loopy graphical model with threshold $r = 9$, the topic number corresponds to the labels of hidden variables in the loopy graph shown in Figure 5.

Graph Structure for Newsgroup data. We employ our method to learn the graph structures under different thresholds $r \in \{3, 5, 7, 9, 11, 13\}$ on newsgroup data, which controls the length of cycles. At $r = 13$ as shown in Fig 6, we obtain a latent tree and for $r \in \{3, 5, 7, 9\}$, we obtain loopy models. The first long cycle appears at $r = 9$ shown in Fig 5. At $r = 7$, we find a combination of short and long cycles. We find that models with cycles are more effective in discovering intuitive relationships. For instance, in the latent tree ($r = 13$), the link between "computer" and "software" is missing due to the tree constraint, but is discovered when $r \leq 9$. Moreover, we see that common words across different topics tend to connect the local subgraphs. For instance, the word "program" is used in the context of both space program and computer programs. Similarly, the word "earth" is used both in the context of religion and space exploration.

Perplexity and Topic Coherence for Newsgroup Data. In Table 1, we present results under our method and under LDA modeling on newsgroup data. For the LDA model, we vary the number of hidden variables (i.e., topics) as $\{10, 20, 30, 40\}$. In contrast, our method is designed to optimize for the number of hidden variables, and does not need this input. We note that our method is competitive in terms of both predictive perplexity and topic coherence. We find that the topic coherence (i.e., PMI) for our method is optimal at $r = 9$, where the graph has a single long cycle and a few short cycles. Intuitively, this model is able to discover more relationships between words, which the latent tree ($r = 13$) is unable to do

Topic 4	Topic 8	Topic 7	Topic 6	Topic 5
Space	windows	card	god	drive
nasa	files	graphics	world	states
insurance	dos	video	fact	research
earth	format	driver	christian	disk
moon	ftp	windows	jesus	university
orbit	program	computer	religion	mac
mission	software	pc	bible	scsi
launch	win	version	evidence	computer
gun	version	software	human	system
shuttle	pc	system	question	power

TABLE 3

Top 10 topic words corresponding to selected topics from the LDA model with 10 topics.

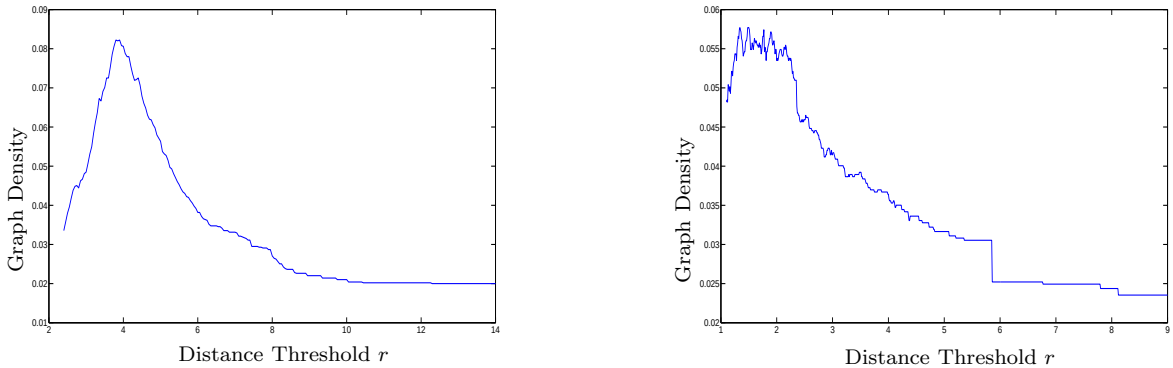
r	Hidden	Edges	Perp-LL	Perp-BIC
2.7	35	154	1.9498	2.0296
3.9	39	139	2.0200	2.0993
4.9	35	129	2.0210	2.0960
5	36	131	2.0169	2.0927
6.7	26	111	2.0344	2.1016
7.7	26	111	2.0353	2.1025
8.2	26	110	2.0405	2.1076

TABLE 4

Comparison of proposed method under different thresholds (r) on Stock data using the proposed method. For definition of perplexity based on test likelihood and BIC scores, see (32), and (34).

so. On the other hand, for $r < 9$, topic coherence is degraded which suggests that adding too many cycles is counterproductive. However, the model at $r = 5$ performs better in terms of predictive perplexity indicating that it is able to use evidence from more observed words for prediction on test data. Moreover, all of our latent graphical models outperform the LDA models in terms of predictive perplexity. The top 10 topic words for selected topics are given for our method at ($r = 9$) and for the LDA model (with 10 topics) are given in Table 2 and Table 3.

Graph Structure for Stock Market Data. The outcome of applying the proposed algorithm to stock market data is presented in Table 4. We observe that the number of edges and hidden variables remain fairly constant over a large range of thresholds. Specifically for $r \in [5.9, 6.7] \cup [6.8, 7.7]$, we obtain the same graph structure (for $r > r_{\max}$, we obtain a tree). Another general trend observed is the improvement of the BIC score as the threshold decreases up till a certain point. The graphs learned using $r = 5$, 7.7 and 8.2 are shown in Fig.7, Fig.8, and Fig.9. Interesting connections between companies emerge. The latent tree structure in Fig.9 captures many key relationships. In particular, the S&P index node has a high degree since it captures the overall trend of various companies. Companies in similar sectors and divisions are grouped together. For instance, retail stores such as "Target", "Walmart", "CVS" and "Home Depot" are grouped together. However, additional relationships emerge as the threshold is decreased and cycles are added. We observe that the first cycle that is added connects the various oil companies which suggests strong interdependencies and influence on the S&P100 index. In addition, more cycles emerge when the threshold is decreased further. For instance, in Figure 7, we find a cycle connecting aviation company "Boeing" with "Honeywell" which is in aviation industry, but also additionally is in chemical industry



(a) Newsgroup data. (b) Stock data.

FIG 4. *Variation of edge density of graphs at the initialization stage of* LocalCLGrouping *vs. threshold* r.

and connects to companies such as "Dow Chemicals". Thus, as in newsgroup data, we find that companies in multiple categories lead to cycles in the underlying graph.

Edge Density vs. Threshold r. We now study the edge density (i.e, number of edges) in the initialization step of our method as a function of the threshold r for both newsgroup and stock data. Recall that the initialization step involves building a loopy graph on observed variables (and no hidden variables). The edge density in this step is indicative of number of cycles added to the ultimate latent model. We observe that the graphs become denser as r is reduced from $r_{\max}$. However, when r is very small, the number of edges decreases since the nodes have sparser neighborhoods. This trend is seen in both : Figures 4a and 4b show the variation for newsgroup and stock data. For the newsgroup data, the graph density peaks at $r = 5$, which also achieves the highest predictive perplexity (see Table 1). Thus, we see a direct relationship between the edge density and the corresponding predictive perplexity in the learnt model. Intuitively, this is because as the number of edges increases, prediction at any node involves more evidence. However, as the threshold r is reduced further, graphs become less denser, and there is also a corresponding degradation in the predictive perplexity.

The above experiments confirm the effectiveness of our approach for discovering hidden topics, and are in line with the theoretical guarantees established earlier in the paper. Our analysis reveals that a large class of loopy graphical models with latent variables can be learnt efficiently in different domains.

7. Conclusion. In this paper, we considered latent graphical models Markov on girth-constrained graphs and proposed a novel approach for structure estimation. We established the correctness of the method when the model is in the regime of correlation decay and also derived PAC learning guarantees. We compared these guarantees with other methods for graphical model selection, where there are no latent variables. Our findings reveal that latent variables do not add much complexity to the learning process in certain models and regimes, even when the number of hidden variables is large. These findings push the realm of tractable latent models for learning.

Acknowledgement. The authors thank E. Mossel (Berkeley) for detailed discussions in the beginning regarding problem formulation, modeling and algorithmic approaches, and Padhraic Smyth (UCI) and David Newman (UCI) for evaluation measures for topic models. The authors also thank the editor Tony Cai (Wharton) and anonymous reviewers whose comments substantially improved the paper. An abridged version of this work appears in the Proceedings of NIPS 2012.

References.

[1] Anandkumar, A., Chaudhuri, K., Hsu, D., Kakade, S. M., Song, L. and Zhang, T. (2011). Spectral Methods for Learning Multivariate Latent Tree Structure. *Preprint, ArXiv 1107.1283*.

[2] Anandkumar, A., Tan, V. Y. F., Huang, F. and Willsky, A. S. (2012a). High-Dimensional Gaussian Graphical Model Selection: Walk-Summability and Local Separation Criterion. *J. Machine Learning Research* **13** 2293–2337.

[3] Anandkumar, A., Tan, V. Y. F., Huang, F. and Willsky, A. S. (2012b). High-dimensional structure learning of Ising models: local separation criterion. *The Annals of Statistics* **40** 1346–1375.

[4] Bayati, M., Montanari, A. and Saberi, A. (2009). Generating random graphs with large girth. In *Proc. of ACM-SIAM Symposium on Discrete Algorithms (SODA)*.

[5] Bento, J. and Montanari, A. (2009). Which Graphical Models are Difficult to Learn? In *Proc. of Neural Information Processing Systems (NIPS)*.

[6] Blei, D. M., Ng, A. Y. and Jordan, M. I. (2003). Latent dirichlet allocation. *J. of Machine Learning Research* **3** 993–1022.

[7] Bogdanov, A., Mossel, E. and Vadhan, S. (2008). The complexity of distinguishing Markov random fields. *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques* 331–342.

[8] Bresler, G., Mossel, E. and Sly, A. (2008). Reconstruction of Markov random fields from samples: some observations and algorithms. In *Intl. workshop APPROX Approximation, Randomization and Combinatorial Optimization* 343–356. Springer.

[9] Buneman, P. (1971). The recovery of trees from measures of dissimilarity. *, Mathematics in the Archaeological and Historical Sciences (FR Hodson, DG Kendall, and P. Tautu, eds.)*.

[10] Bunke, H. and Allermann, G. (1983). Inexact graph matching for structural pattern recognition. *Pattern Recognition Letters* **1** 245–253.

[11] Chandran, L. S. and Subramanian, C. (2005). Girth and treewidth. *J. of combinatorial theory, Series B* **93** 23–32.

[12] Chandrasekaran, V., Parrilo, P. A. and Willsky, A. S. (2010). Latent Variable Graphical Model Selection via Convex Optimization. *Arxiv preprint*.

[13] Chandrasekaran, V., Parrilo, P. A. and Willsky, A. S. (2012). Latent Variable Graphical Model Selection via Convex Optimization. *To appear in Annals of Statistics*.

[14] Chen, T., Zhang, N. L. and Wang, Y. (2008). Efficient model evaluation in the search based approach to latent structure discovery. In *4th European Workshop on Probabilistic Graphical Models*.

[15] Choi, M. J., Lim, J. J., Torralba, A. and Willsky, A. S. (2010). Exploiting Hierarchical Context on a Large Database of Object Categories. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[16] Choi, M. J., Tan, V. Y. F., Anandkumar, A. and Willsky, A. (2011). Learning latent tree graphical models. *J. of Machine Learning Research* **12** 1771–1812.

[17] Chung, F. R. K. (1997). *Spectral graph theory*. Amer Mathematical Society.

[18] Daskalakis, C., Mossel, E. and Roch, S. (2006). Optimal phylogenetic reconstruction. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing* 159–168.

[19] Dembo, A. and Montanari, A. (2010). Ising Models on Locally Tree-like Graphs. *Annals of Applied Probability*.

[20] Durbin, R., Eddy, S. R., Krogh, A. and Mitchison, G. (1999). *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge Univ. Press.

[21] Elidan, G. and Friedman, N. (2005). Learning Hidden Variable Networks: The Information Bottleneck Approach. *Journal of Machine Learning Research* **6** 81-127.

[22] Erdös, P. L., Székely, L. A., Steel, M. A. and Warnow, T. J. (1999). A few logs suffice to build (almost) all trees: Part I. *Random Structures and Algorithms* **14** 153–184.

[23] Frank, A. and Asuncion, A. (2010). UCI Machine Learning Repository.

[24] Gamburd, A., Hoory, S., Shahshahani, M., Shalev, A. and Virag, B. (2009). On the girth of random Cayley graphs. *Random Structures & Algorithms* **35** 100–117.

[25] Georgii, H. O. (1988). *Gibbs Measures and Phase Transitions*. Walter de Gruyter.

[26] Jalali, A., Johnson, C. and Ravikumar, P. (2011). On Learning Discrete Graphical Models Using Greedy Methods. In *Proc. of NIPS*.

[27] Karger, D. and Srebro, N. (2001). Learning Markov networks: maximum bounded tree-width graphs. In *Proc. of ACM-SIAM Symposium on Discrete algorithms* 392–401.

[28] Kearns, M. J. and Vazirani, U. V. (1994). *An Introduction to Computational Learning Theory*. MIT Press., Cambridge, MA.

[29] Kemp, C. and Tenenbaum, J. B. (2008). The discovery of structural form. *Proceedings of the National Academy of Science* **105** 10687-10692.

[30] Lauritzen, S. L. (1996). *Graphical models*. Clarendon Press.

[31] Lazarsfeld, P. F. and Henry, N. W. (1968). *Latent structure analysis*. Boston: Houghton Mifflin.

[32] Meinshausen, N. and Bühlmann, P. (2006). High dimensional graphs and variable selection with the lasso. *Annals of Statistics* **34** 1436–1462.

[33] Mezard, M. and Montanari, A. (2009). *Information, physics, and computation*. Oxford University Press, USA.

[34] Mossel, E. (2007). Distorted metrics on trees and phylogenetic forests. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 108–116.

[35] Mossel, E. and Roch, S. (2006). Learning nonsingular phylogenies and hidden Markov models. *The Annals of Applied Probability* **16** 583–614.

[36] Netrapalli, P., Banerjee, S., Sanghavi, S. and Shakkottai, S. (2010). Greedy learning of Markov network structure . In *Proc. of Allerton Conf. on Communication, Control and Computing*.

[37] Newman, D., Bonilla, E. V. and Buntine, W. (2011). Improving Topic Coherence with Regularized Topic Models. In *Proc. of NIPS*.

[38] Newman, D., Karimi, S. and Cavedon, L. (2009). External Evaluation of Topic Models. In *Proceedings of the 14th Australasian Computing Symposium(ACD2009)* 8.

[39] Ravikumar, P., Wainwright, M. J. and Lafferty, J. (2008). High-dimensional Ising Model Selection Using l1-Regularized Logistic Regression. *Annals of Statistics*.

[40] Ravikumar, P., Wainwright, M. J., Raskutti, G. and Yu, B. (2011). High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics* 4 935–980.

[41] Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics* **6** 461–464.

[42] Semple, C. and Steel, M. (2003). *Phylogenetics*. Oxford University Press.

[43] Song, L., Parikh, A. P. and Xing, E. P. (2011). Kernel Embeddings of Latent Tree Graphical Models. In *Proc. of NIPS*.

[44] Steel, M. (1994). Recovering a tree from the leaf colourations it generates under a Markov model. *Applied Mathematics Letters* **7** 19–23.

[45] Weitz, D. (2005). Combinatorial Criteria for Uniqueness of Gibbs Measures. *Random Structures & Algorithms* **27** 445.

[46] Zhang, N. L. (2004). Hierarchical Latent Class Models for Cluster Analysis. *Journal of Machine Learning Research* **5** 697–723.

[47] Zhang, N. L. and Kocka, T. (2004). Efficient Learning of Hierarchical Latent Class Models. In *ICTAI*.

Electrical Engineering & Computer Science Dept.,
4408 Engineering Hall, Irvine, CA, USA 92697.
E-mail: a.anandkumar@uci.edu; rvalluva@uci.edu

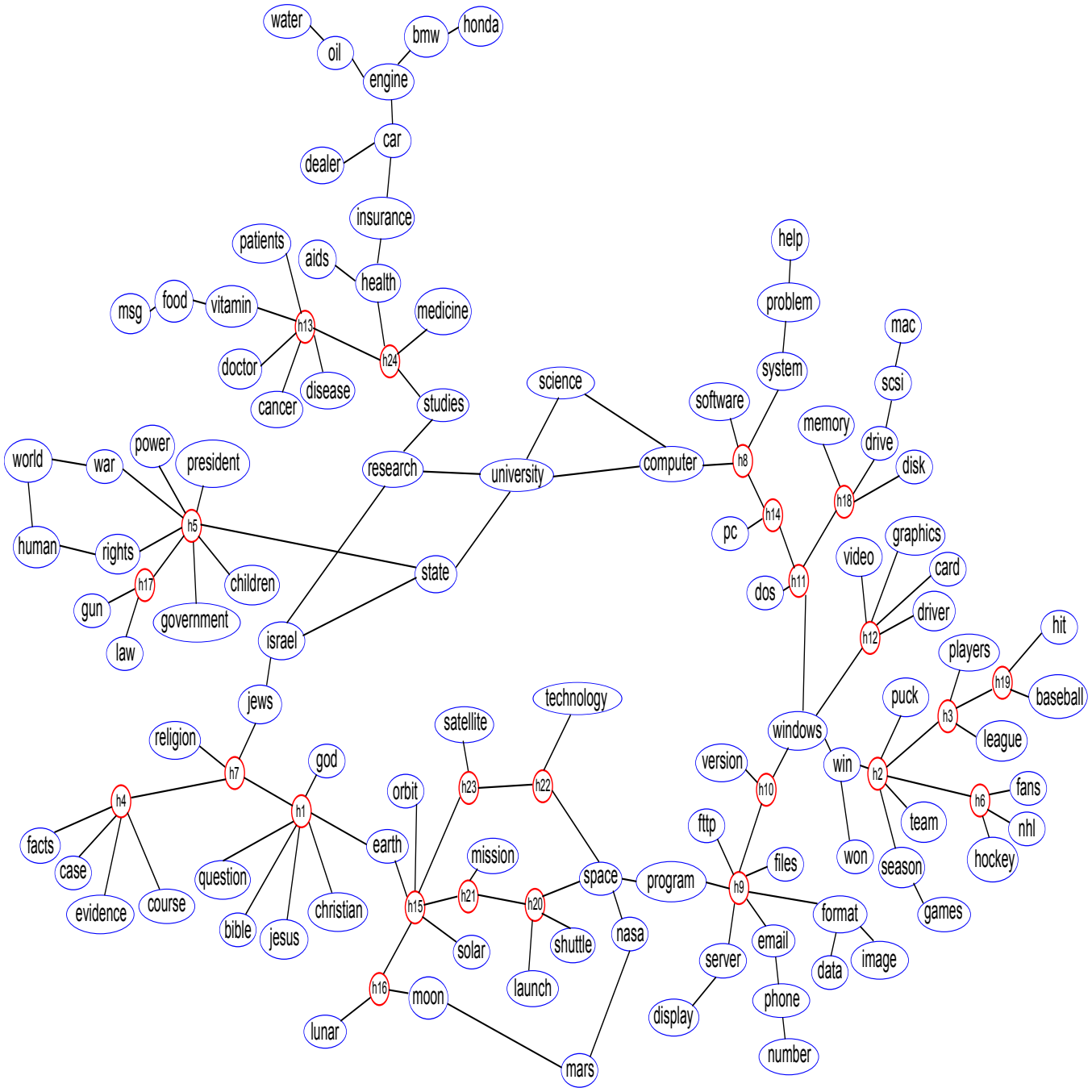


FIG 5. *Loopy Graph Learned using* $r = 9$ *with* RegLocalCLGrouping *on 20 newsgroup data.*

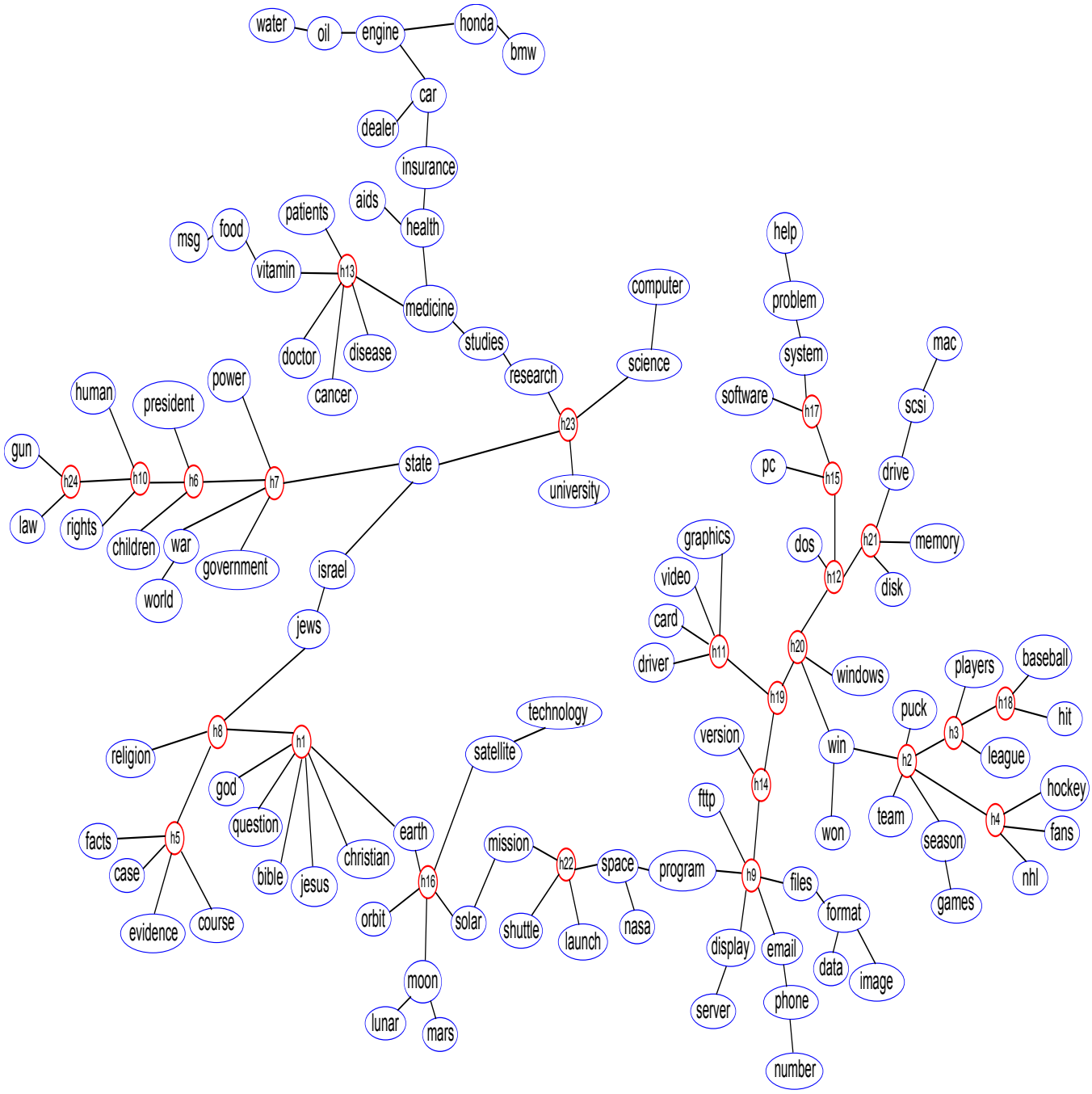


FIG 6. *Tree Graph Learned using* $r = 13$ *with* RegLocalCLGrouping *on 20 newsgroup data.*

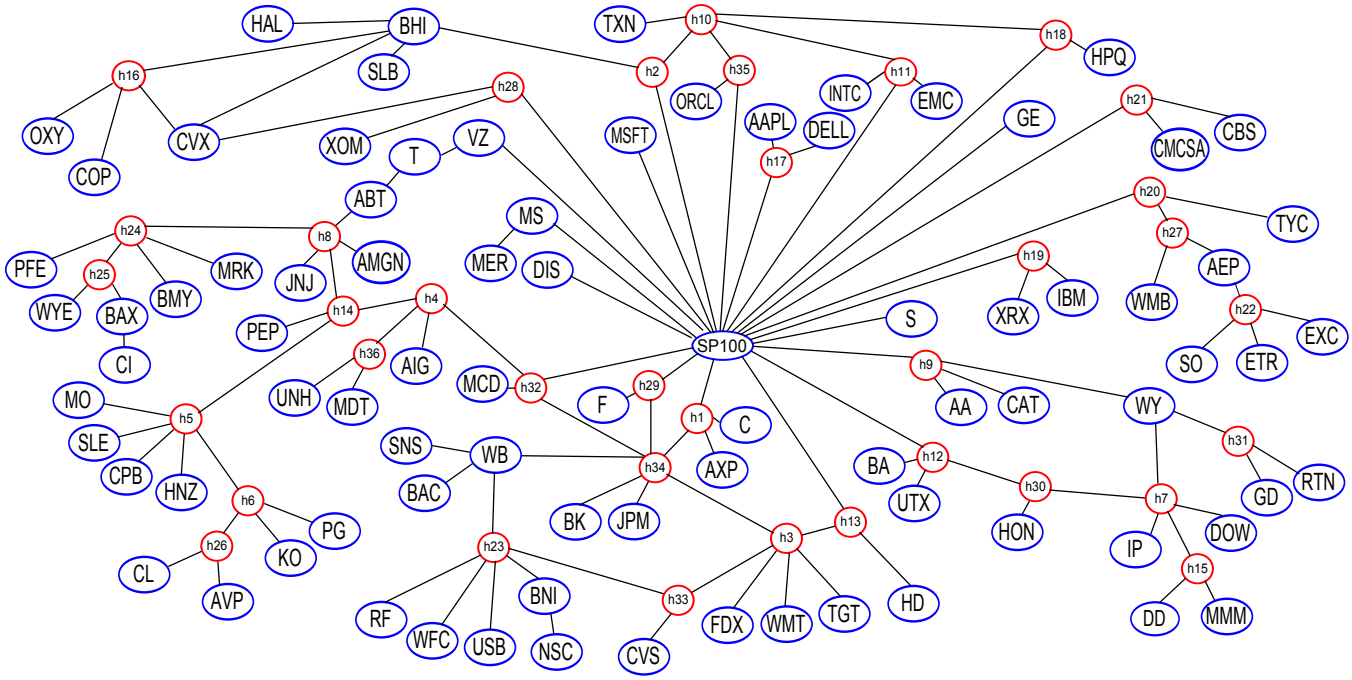


FIG 7. *Loopy Graph Learned using* $r = 5$ *with* LocalCLGrouping *on S&P 100 monthly stock return data.*

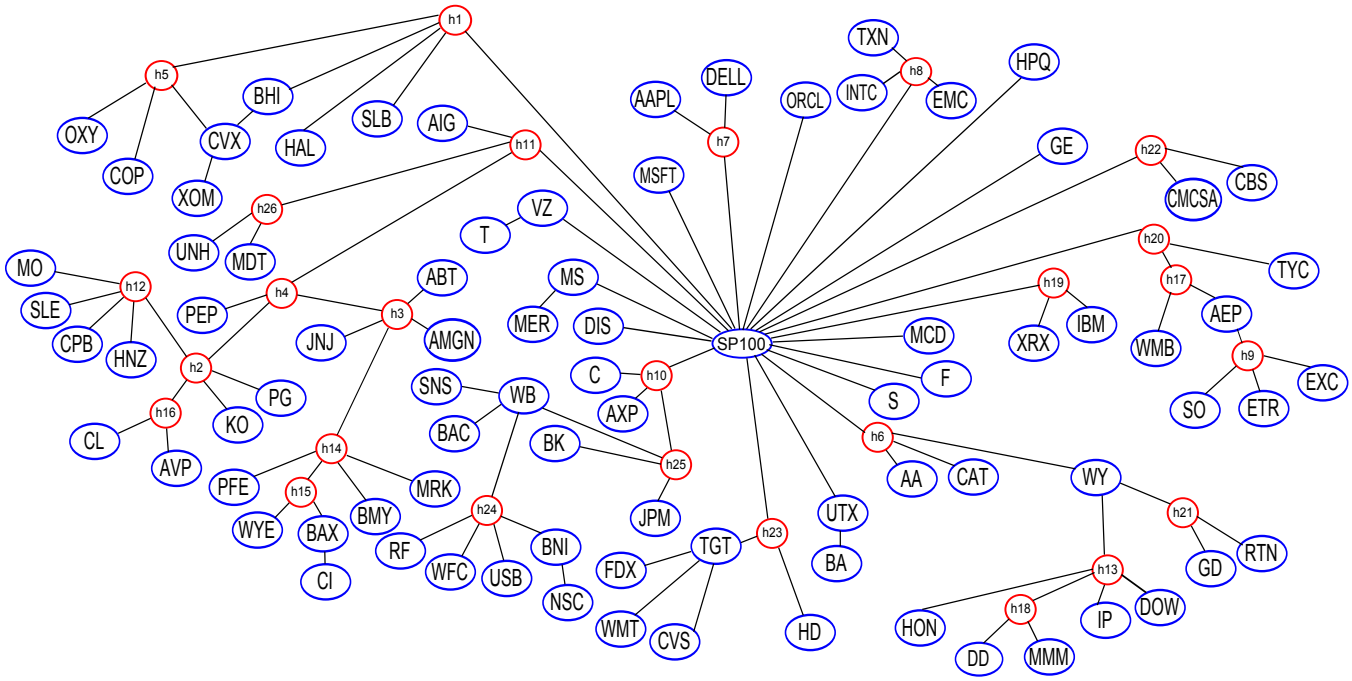


FIG 8. *Loopy Graph Learned using* $r = 7.7$ *with* LocalCLGrouping *on S&P 100 monthly stock return data.*

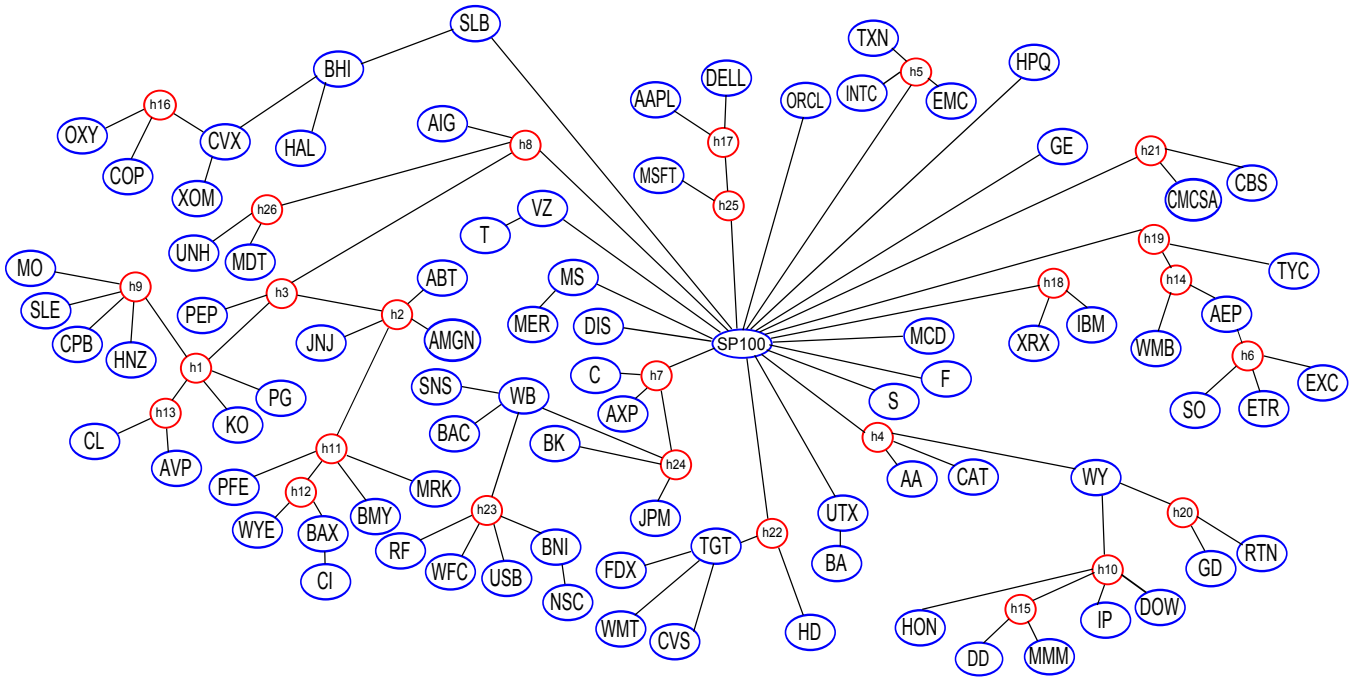


FIG 9. *Tree Graph Learned using* $r = 8.2$ *with* LocalCLGrouping *on S&P 100 monthly stock return data.*