Learning Planar Ising Models

Jason K. Johnson   JASON.JOHNSON@NUMERICA.US
Numerica
Ft. Collins, CO, USA

Diane Oyen   DOYEN@LANL.GOV
Michael Chertkov   CHERTKOV@LANL.GOV
Los Alamos National Laboratory
Los Alamos, NM, USA

Praneeth Netrapalli   PRANEETH@MICROSOFT.COM
Microsoft Research
Cambridge, MA, USA

Abstract

Inference and learning of graphical models are both well-studied problems in statistics and machine learning that have found many applications in science and engineering. However, exact inference is intractable in general graphical models, which suggests the problem of seeking the best approximation to a collection of random variables within some tractable family of graphical models. In this paper, we focus on the class of planar Ising models, for which exact inference is tractable using techniques of statistical physics. Based on these techniques and recent methods for planarity testing and planar embedding, we propose a greedy algorithm for learning the best planar Ising model to approximate an arbitrary collection of binary random variables (possibly from sample data). Given the set of all pairwise correlations among variables, we select a planar graph and optimal planar Ising model defined on this graph to best approximate that set of correlations. We demonstrate our method in simulations and for two applications: modeling senate voting records and identifying geo-chemical depth trends from Mars rover data.

Keywords: Ising models, graphical models

1. Introduction

Graphical models are widely used to represent the statistical relations among a set of random variables (Lauritzen, 1996; MacKay, 2003). Nodes of the graph correspond to random variables and edges of the graph represent statistical interactions among the variables. The problems of inference and learning on graphical models arise in many practical applications. The problem of inference is to deduce certain statistical properties (such as marginal probabilities, modes etc.) of a given set of random variables whose graphical model is known. Inference has wide applications in areas such as error correcting codes, statistical physics and so on. The problem of learning on the other hand is to deduce the graphical model of a set of random variables given statistics (possibly from samples) of the random variables. Learning is also a widely encountered problem in areas such as biology, neuroscience and so on (Barabasi and Oltvai, 2004; Smith et al., 2011).

The *Ising model*, a class of binary-variable graphical models with pairwise interactions, has been studied by physicists as a simple model of order-disorder transitions in magnetic materials (Onsager, 1944). Remarkably, it was found that in the special case of an Ising model with zero-mean $\{-1, +1\}$ binary random variables and pairwise interactions defined on a planar graph, calculation of the partition function (which is closely tied to inference) is tractable, essentially reducing to calculation of a matrix determinant (Kac and Ward, 1952; Sherman, 1960; Kasteleyn, 1963; Fisher, 1966). Planar graph inference methods have been used in machine learning for efficient inference on planar graphs (Schraudolph and Kamenetsky, 2008; Gómez et al., 2010), in approximating inference for general graphs with planar graph decomposition (Jaakkola and T., 2007); and applied to problems such as computer vision (Batra et al., 2010; Yarkony et al., 2012) and financial forecasting (Pozzi et al., 2013), usually as an approximation method for problems with non-binary data.

We address the problem of approximating a collection of binary random variables (given their pairwise marginal distributions) by a zero-mean planar Ising model. We also consider the related problem of selecting a non-zero mean Ising model defined on an outer-planar graph (these models are also tractable, being essentially equivalent to a zero-field model on a related planar graph).

There has been a great deal of work on learning graphical models. Much of these have focused on learning over the class of thin graphical models (Deshpande et al., 2001; Bach and Jordan, 2001; Karger and Srebro, 2001; Shahaf et al., 2009) for which inference is tractable by converting the model to a junction tree. The simplest case of this is learning tree models (treewidth one graphs) for which it is tractable to find the best tree model by reduction to a max-weight spanning tree problem (Chow and Liu, 1968). However, the problem of finding the best bounded-treewidth model is NP-hard for treewidths greater than two (Karger and Srebro, 2001), and so heuristic methods are used to select the graph structure (Deshpande et al., 2001; Karger and Srebro, 2001). Another popular method is to use convex optimization of the log-likelihood penalized by the ℓ_1 norm of parameters of the graphical model so as to promote sparsity (Banerjee et al., 2008; Lee et al., 2006). To go beyond low-treewidth graphs, such methods either focus on Gaussian graphical models or adopt a tractable approximation of the likelihood. Other methods learn only the graph structure itself (Ravikumar et al., 2010; Abbeel et al., 2006) and are often able to demonstrate asymptotic correctness of this estimate under appropriate conditions.

In contrast to existing approaches, this paper explores planarity as an alternative restriction on the model class, instead of low treewidth, to make learning tractable while maintaining a tractable model for inference, unlike unrestricted regularized approaches, while still providing a solution in which the number of edges learned is linear in the number of variables.

2. Preliminaries

We develop our notation and briefly review the necessary background theory on graph estimation, Ising models and exact inference in planar graphs.

2.1 Divergence and Likelihood

A graph represents a joint probability distribution over a collection of variables. Suppose we want to calculate how well a probability distribution Q approximates another probability distribution P (on the same sample space χ). For any two probability distributions P and Q on some sample space χ, we denote by $D(P, Q)$ the *Kullback-Leibler divergence* (or *relative entropy*) between P and Q as $D(P, Q) = \sum_{x \in \chi} P(x) \log \frac{P(x)}{Q(x)}$. The *log-likelihood function* is defined as $LL(P, Q) = \sum_{x \in \chi} P(x) \log Q(x)$. The probability distribution in a family $\mathcal{F}$ that maximizes the log-likelihood of a probability distribution P is called the *maximum-likelihood estimate* of P in $\mathcal{F}$, and this is equivalent to the *minimum-divergence projection* of P to $\mathcal{F}$, so that $P_{\mathcal{F}} = \arg\max_{Q \in \mathcal{F}} LL(P, Q) = \arg\min_{Q \in \mathcal{F}} D(P, Q)$.

2.2 Graphical Models and The Ising Model

We will be dealing with binary random variables throughout the paper. We write $P(x)$ to denote the probability distribution of a collection of random variables $x = (x_1, \ldots, x_n)$. Unless otherwise stated, we work with undirected graphs $G = (V, E)$ with vertex (or node) set V and edges $\{i, j\} \in E \subset \binom{V}{2}$. For vertices $i, j \in V$ we write $G + ij$ to denote the graph $(V, E \cup \{i, j\})$. A *pairwise graphical model* is a probability distribution $P(x) = P(x_1, \ldots, x_n)$ that is defined on a graph $G = (V, E)$ with vertices $V = \{1, .., n\}$ as

$$
\begin{aligned}
P(x) &\propto \prod_{i \in V} \psi_i(x_i) \prod_{\{i,j\} \in E} \psi_{ij}(x_i, x_j) \\
&\propto \exp\left\{ \sum_{i \in V} f_i(x_i) + \sum_{\{i,j\} \in E} f_{ij}(x_i, x_j) \right\},
\end{aligned}
\tag{1}
$$

where $\psi_i, \psi_{ij} \geq 0$ are non-negative node and edge compatibility functions. For positive ψ's, we may also represent $P(x)$ as a Gibbs distribution with potentials $f_i = \log \psi_i$ and $f_{ij} = \log \psi_{ij}$.

Definition 1 *An* Ising model *on binary random variables $x = (x_1, \ldots, x_n)$ and graph $G = (V, E)$ is the probability distribution defined by*

$$
P(x) = \frac{1}{Z(\theta)} \exp\left\{ \sum_{i \in V} \theta_i x_i + \sum_{\{i,j\} \in E} \theta_{ij} x_i x_j \right\},
$$

$$
Z(\theta) = \sum_x \exp\left\{ \sum_{i \in V} \theta_i x_i + \sum_{\{i,j\} \in E} \theta_{ij} x_i x_j \right\},
$$

where $x_i \in \{-1, 1\}$. The partition function $Z(\theta)$ *serves to normalize the probability distribution.*

Formally, this defines an *exponential family* $P_\theta(x) = \exp\{\theta^T \phi(x) - \Phi(\theta)\}$ (Barndorff-Nielsen, 1979; Wainwright and Jordan, 2008) based on sufficient statistics $(\phi_i(x) = x_i, i \in V)$ and $(\phi_{ij}(x) = x_i x_j, \{i, j\} \in E)$, parameters $(\theta_i, i \in V)$ and $(\theta_{ij}, \{i, j\} \in E)$ and moment

parameters ($\mu_i = \mathbb{E}[x_i], i \in V$) and ($\mu_{ij} = \mathbb{E}[x_i x_j], \{i,j\} \in E$). The function $\Phi(\theta) = \log Z(\theta)$ is a convex function of θ and has the moment generating properties: $\nabla \Phi(\theta) = \mathbb{E}_\theta[\phi(x)] = \mu$ and $\nabla^2 \Phi(\theta) = \mathbb{E}_\theta[(\phi(x) - \mu)(\phi(x) - \mu)^T]$.

In fact, any pairwise graphical model among binary variables can be represented as an Ising model:

$$\theta_i = \tfrac{1}{2} \sum_{x_i} x_i f_i(x_i) + \tfrac{1}{4} \sum_{\{i,j\} \in E} \sum_{x_i,x_j} x_i f_{ij}(x_i, x_j),$$

$$\theta_{ij} = \tfrac{1}{4} \sum_{x_i,x_j} x_i x_j f_{ij}(x_i, x_j).$$

The moments can be computed as: $\mu_i = \sum_{x_i} x_i P(x_i)$ and $\mu_{ij} = \sum_{x_i,x_j} x_i x_j P(x_i, x_j)$. Inversely, the marginals are computed by:

$$P(x_i) = \tfrac{1}{2}(1 + \mu_i x_i),$$

$$P(x_i, x_j) = \tfrac{1}{4}(1 + \mu_i x_i + \mu_j x_j + \mu_{ij} x_i x_j).$$

We will be especially concerned with the following sub-family of Ising models:

Definition 2 *An Ising model is said to be* zero-field *if $\theta_i = 0$ for all $i \in V$. It is* zero-mean *if $\mu_i = 0$ ($P(x_i = \pm 1) = \tfrac{1}{2}$) for all $i \in V$.*

The Ising model is zero-field if and only if it is zero-mean. Although the zero-field assumption appears very restrictive, a general Ising model can be represented as a zero-field model by adding one auxiliary variable node connected to every other node of the graph (Jaakkola and T., 2007). The parameters and moments of the two models are then related as follows:

Proposition 1 *Consider the Ising model on $G = (V, E)$ with $V = \{1, \ldots, n\}$, parameters $\{\theta_i\}$ and $\{\theta_{ij}\}$, moments $\{\mu_i\}$ and $\{\mu_{ij}\}$ and partition function Z. Let $\widehat{G} = (\widehat{V}, \widehat{E})$ denote the extended graph based on nodes $\widehat{V} = V \cup \{n+1\}$ with edges $\widehat{E} = E \cup \{\{i, n+1\}, i \in V\})$. We define a zero-field Ising model on $\widehat{G}$ with parameters $\{\widehat{\theta}_{ij}\}$, moments $\{\widehat{\mu}_{ij}\}$ and partition function $\widehat{Z}$. If we set the parameters according to*

$$\widehat{\theta}_{ij} = \begin{cases} \theta_i & \text{if } j = n+1 \\ \theta_{ij} & \text{otherwise} \end{cases},$$

then $\widehat{Z} = 2Z$ and

$$\widehat{\mu}_{ij} = \begin{cases} \mu_i & \text{if } j = n+1 \\ \mu_{ij} & \text{otherwise} \end{cases}.$$

Thus, inference on the corresponding zero-field Ising model on the extended graph $\widehat{G}$ is equivalent to inference on the (non-zero-field) Ising model defined on G. Proof given in Appendix A.

2.3 Inference for Planar Ising Models

The motivation for our paper is the following result on tractability of inference for the *planar zero-field Ising model*.

Definition 3 *A graph is* planar *if it may be embedded in the plane without any edge crossings.*

Moreover, it is known that any planar graph can be embedded such that all edges are drawn as straight lines.

Theorem 1 *(Kac and Ward, 1952; Sherman, 1960; Loebl, 2010) Let G be a planar graph with specified straight-line embedding in the plane and let $\phi_{ijk} \in [-\pi, +\pi]$ denote the clockwise rotation between the directed edges (i, j) and (j, k). We define the matrix $W \in \mathbb{C}^{2|E| \times 2|E|}$ indexed by directed edges of the graph as follows: $W = AD$ where D is the diagonal matrix with $D_{ij,ij} = \tanh \theta_{ij} \triangleq w_{ij}$ and*

$$A_{ij,kl} = \begin{cases} \exp(\frac{1}{2}\sqrt{-1}\phi_{ijl}), & j = k \text{ and } i \neq l \\ 0, & \text{otherwise.} \end{cases}$$

Then, the partition function of the zero-field planar Ising model is given by the Kac-Ward determinant formula:

$$Z = 2^n \left(\prod_{\{i,j\} \in E} \cosh \theta_{ij} \right) \det(I - W)^{\frac{1}{2}} \quad .$$

Another related method for computing the Ising model partition function is based on counting perfect matchings of planar graphs (Kasteleyn, 1963; Fisher, 1966). Thus, calculating the partition function reduces to calculating the determinant of a matrix; therefore, using the generalized nested dissection algorithm to exploit sparsity of the matrix, the complexity of these calculations is $O(n^{3/2})$ (Lipton et al., 1979; Lipton and Tarjan, 1979; Galluccio et al., 2000). Thus, inference of the zero-field planar Ising model is tractable and scales well with problem size.

The gradient and Hessian of the log-partition function $\Phi(\theta) = \log Z(\theta)$ can also be calculated efficiently from the Kac-Ward determinant formula. Derivatives of $\Phi(\theta)$ recover the moment parameters of the exponential family model as $\nabla \Phi(\theta) = \mathbb{E}_\theta[\phi] = \mu$ (Barndorff-Nielsen, 1979; Wainwright and Jordan, 2008). Thus, inference of moments (and node and edge marginals) is tractable for the zero-field planar Ising model.

Proposition 2 *Let $\mu = \nabla \Phi(\theta)$, $H = \nabla^2 \Phi(\theta)$. Let $S = (I - W)^{-1}A$ and $T = (I + P)(S \circ S^T)(I + P^T)$ where A and W are defined as in Theorem 1, $\circ$ denotes the element-wise product and P is the permutation matrix swapping the indices of the directed edges (i, j) and (j, i). Then,*

$$\mu_{ij} = w_{ij} - \frac{1}{2}(1 - w_{ij}^2)(S_{ij,ij} + S_{ji,ji})$$

$$H_{ij,kl} = \begin{cases} 1 - \mu_{ij}^2, & ij = kl \\ -\frac{1}{2}(1 - w_{ij}^2)T_{ij,kl}(1 - w_{kl}^2), & \text{otherwise.} \end{cases}$$

5

Calculating the full matrix S requires $O(n^3)$ calculations. However, to compute just the moments μ only the diagonal elements of S are needed. Then, using the generalized nested dissection method, inference of moments (edge-wise marginals) of the zero-field Ising model can be achieved with complexity $O(n^{3/2})$. Computing the full Hessian is more expensive, requiring $O(n^3)$ calculations.

2.3.1 INFERENCE FOR OUTER-PLANAR GRAPHICAL MODELS

We emphasize that the above calculations require both a planar graph G and a zero-field Ising model. Using the graphical transformation of Proposition 1, the latter zero-field condition may be relaxed but at the expense of adding an auxiliary node connected to all the other nodes. In general planar graphs G, the new graph $\widehat{G}$ may not be planar and hence may not admit tractable inference calculations. However, for the subset of planar graphs where this transformation does preserve planarity inference is still tractable.

Definition 4 *A graph G is said to be* outer-planar *if there exists an embedding of G in the plane where all the nodes are on the outer face.*

In other words, the graph G is outer-planar if the extended graph $\widehat{G}$ (defined by Proposition 1) is planar. Then, from Proposition 1 and Theorem 1 it follows that:

Proposition 3 *(Jaakkola and T., 2007) The partition function and moments of any outer-planar Ising graphical model (not necessarily zero-field) can be calculated efficiently. Hence, inference is tractable for any binary-variable graphical model with pairwise interactions defined on an outer-planar graph.*

This motivates the problem of learning outer-planar graphical models for a collection of (possibly non-zero mean) binary random variables.

3. Learning Planar Ising Models

This section addresses the main goals of the paper, which are two-fold:

1. Solving for the maximum-likelihood Ising model on a given planar graph to best approximate a collection of zero-mean random variables.

2. Selecting heuristically the planar graph to obtain the best approximation.

We address these problems in the following two subsections. The solution of the first problem is an integral part of our approach to the second. Both solutions are easily adapted to the context of learning outer-planar graphical models of (possibly non-zero mean) binary random variables.

3.1 Maximum-Likelihood Parameter Estimation

Maximum-likelihood estimation over an exponential family is a convex optimization problem based on the log-partition function $\Phi(\theta)$. In the case of the zero-field Ising model defined on a given planar graph it is tractable to compute $\Phi(\theta)$ via a matrix determinant described in

Theorem 1. Thus, we obtain an unconstrained, tractable, convex optimization problem for the maximum-likelihood zero-field Ising model on the planar graph G to best approximate a probability distribution $P(x)$:

$$\max_{\theta}\{\mu^T\theta - \Phi(\theta)\} = \max_{\theta \in \mathbb{R}^{|E|}} \left\{ \sum_{ij}(\mu_{ij}\theta_{ij} - \log\cosh\theta_{ij}) - \tfrac{1}{2}\log\det(I - W(\theta)) \right\}.$$

Here, $\mu_{ij} = \mathbb{E}_P[x_ix_j]$ for all edges $\{i,j\} \in G$ and the matrix $W(\theta)$ is as defined in Theorem 1. If P represents the empirical distribution of a set of independent identically-distributed (iid) samples $\{x^{(s)}, s = 1, \ldots, S\}$ then $\{\mu_{ij}\}$ are the corresponding empirical moments $\mu_{ij} = \frac{1}{S}\sum_s x_i^{(s)}x_j^{(s)}$.

3.1.1 NEWTON'S METHOD

We solve this unconstrained convex optimization problem using Newton's method with step-size chosen by back-tracking line search (Boyd and Vandenberghe, 2004). This produces a sequence of estimates $\theta^{(t)}$ calculated as follows:

$$\theta^{(t+1)} = \theta^{(t)} + \lambda_t H(\theta^{(t)})^{-1}(\mu(\theta^{(t)}) - \mu),$$

where $\mu(\theta^{(t)})$ and $H(\theta^{(t)})$ are calculated using Proposition 2 and $\lambda_t \in (0,1]$ is a step-size parameter chosen by backtracking line search (see Boyd and Vandenberghe (2004): Chapter 9, Section 2 for details). The per iteration complexity of this optimization is $O(n^3)$ using explicit computation of the Hessian at each iteration. This complexity can be offset somewhat by only re-computing the Hessian a few times (reusing the same Hessian for a number of iterations), to take advantage of the fact that the gradient computation only requires $O(n^{\frac{3}{2}})$ calculations. As Newton's method has quadratic convergence, the number of iterations required to achieve a high-accuracy solution is typically 8-16 iterations (essentially independent of problem size). We estimate the computational complexity of solving this convex optimization problem as roughly $O(n^3)$.

3.2 Greedy Planar Graph Selection

We now consider the problem of selection of the planar graph G to best approximate a probability distribution $P(x)$ with pairwise moments $\mu_{ij} = \mathbb{E}_P[x_ix_j]$ given for all $i, j \in V$. Formally, we seek the planar graph that maximizes the log-likelihood (minimizes the divergence) relative to P:

$$\widehat{G} = \underset{G \in \mathcal{P}_V}{\arg\max} LL(P, P_G) = \underset{G \in \mathcal{P}_V}{\arg\max} \max_{Q \in \mathcal{F}_G} LL(P, Q),$$

where $\mathcal{P}_V$ is the set of planar graphs on the vertex set V, $\mathcal{F}_G$ denotes the family of zero-field Ising models defined on graph G and $P_G = \arg\max_{Q \in \mathcal{F}_G} LL(P, Q)$ is the maximum-likelihood (minimum-divergence) approximation to P over this family.

We obtain a heuristic solution to this graph selection problem using the following greedy edge-selection procedure. The input to the algorithm is a probability distribution P (which could be empirical) on n binary $\{-1, 1\}$ random variables. In fact, it is sufficient to summarize P by its pairwise correlations $\mu_{ij} = \mathbb{E}_P[x_ix_j]$ on all pairs $i, j \in V$. The output is a

maximal planar graph G and the maximum-likelihood approximation θ_G to P in the family of zero-field Ising models defined on this graph. Note that a maximal planar graph is a planar graph for which no new edge can be added that would maintain planarity. Planar graphs are inherently sparse. All maximal planar graphs with $n > 2$ have $3n - 6$ edges[1].

Algorithm 1 GreedyPlanarGraphSelect(P)

1: $G = \emptyset, \theta_G = 0$
2: **for** $k = 1 : 3n - 6$ **do**   ▷ Add edges until maximal planar graph reached
3:     $\Delta = \{\{i, j\} \subset V | \{i, j\} \notin G, G + ij \in \mathcal{P}_V\}$ ▷ Set of edges that preserve planarity
4:     $\tilde{\mu}_\Delta = \{\tilde{\mu}_{ij} = \mathbb{E}_{\theta_G}[x_i x_j], \{i, j\} \in \Delta\}$ ▷ Compute pairwise correlations
5:     $G \leftarrow G \cup \arg\max\limits_{e \in \Delta} D(P_e, \tilde{P}_e)$ ▷ Select edge that maximizes gain in log-likelihood
6:     $\theta_G = \text{PlanarIsing}(G, P)$ ▷ Compute maximum-likelihood parameters for G
7: **end for**

The algorithm starts with an empty graph and then sequentially adds edges to the graph one at a time so as to greedily increase the log-likelihood (decrease the divergence) relative to P as much as possible at each step. Here is a more detailed description of the algorithm along with estimates of the computational complexity of each step:

- *Line 3.* First, we enumerate the set Δ of all edges one might add (individually) to the graph while preserving planarity. This is accomplished by an $O(n^3)$ algorithm in which we iterate over all pairs $\{i, j\} \notin G$ and for each such pair we form the graph $G + ij$ and test planarity of this graph using known $O(n)$ algorithms (Chrobak and Payne, 1995).

- *Line 4.* Next, we perform tractable inference calculations with respect to the Ising model on G to calculate the pairwise correlations $\tilde{\mu}_{ij}$ for all pairs $\{i, j\} \in \Delta$. This is accomplished using $O(n^{3/2})$ inference calculations on augmented versions of the graph G. For computational efficiency instead of calculating the addition of each proposed edge individually, moments of proposed edges are calculated in batches as follows. For each inference calculation we add as many edges to G from Δ as possible (setting $\theta = 0$ on these edges) while preserving planarity and then calculate all the edge-wise moments of this graph using Proposition 2 (including the zero-edges). This requires at most $O(n)$ iterations to cover all pairs of Δ, so the worst-case complexity to compute all required pairwise moments is $O(n^{5/2})$.

- *Line 5.* Once we have these moments, which specify the corresponding pairwise marginals of the current Ising model, we compare these moments (pairwise marginals) to those of the input distribution P by evaluating the pairwise KL-divergence between the Ising model and P. As seen by the following proposition, this gives us a lower-

---

1. Euler's formula states that for a planar graph, $n - e + f = 2$, where f is the number of faces and e is the number of edges (Richeson, 2012). The faces of a planar graph are simple polygons, so each face has at least 3 edges and each edge is part of at most 2 faces, $2e \leq 3f$. Combining these two properties, gives $e \leq 3n - 6$ for planar graphs. A maximal planar graph is triangulated, meaning that all faces have three sides, and therefore $e = 3n - 6$ (if any face has more than 3 sides, an edge can be added as a chord of that face polygon without breaking planarity).

bound on the improvement obtained by adding edge $\{i, j\}$ (see Appendix A for proof):

Proposition 4 *Let P_G and P_{G+ij} be projections of P on G and $G + ij$ respectively. Then,*

$$D(P, P_G) - D(P, P_{G+ij}) \geq D\left(P(x_i, x_j), P_G(x_i, x_j)\right),$$

where $P(x_i, x_j)$ and $P_G(x_i, x_j)$ represent the marginal distributions on x_i, x_j of probabilities P and P_G respectively.

Thus, we greedily select the next edge $\{i, j\}$ to add so as to maximize this lower-bound on the improvement measured by the increase on log-likelihood (this being equal to the decrease in KL-divergence).

- *Line 6.* Finally, we calculate the new maximum-likelihood parameters θ_G on the new graph $G \leftarrow G + ij$. This involves solving the convex optimization problem discussed in the preceding subsection, which requires $O(n^3)$ complexity. This step is necessary in order to subsequently calculate the pairwise moments $\tilde{\mu}$ which guide further edge-selection steps, and also to provide the final estimate.

We continue adding one edge at a time until a maximal planar graph (with $3n - 6$ edges) is obtained. Thus, the total complexity of our greedy algorithm for planar graph selection is $O(n^4)$.

3.2.1 Non-Maximal Planar Graphs

Since adding an edge always improves the log-likelihood, the greedy algorithm always outputs a maximal planar graph. However, this might lead to over-fitting of the data especially when the input probability distribution is an empirical distribution. Note that at $3n - 6$ edges, the maximal planar graph is sparse and our empirical work indicates that over-fitting is often not an issue. In the case that over-fitting is a concern, we could terminate the algorithm when adding an edge to the graph would only improve the log-likelihood by less than some threshold γ. A data-driven search can be performed for a suitable value of this threshold so as to minimize some estimate of the generalization, such as in cross validation methods (Zhang, 1993). Or, one could use some heuristic value for γ based on the number of samples such as Akaike's information criterion (AIC) or Schwarz's Bayesian information criterion (BIC) (Akaike, 1974; Schwarz, 1978).

3.2.2 Outer-Planar Graphs and Non-Zero Means

The greedy algorithm returns a zero-field Ising model (which has zero mean for all the random variables) defined on a planar graph. If the actual random variables are non-zero mean, this may not be desirable. For this case we may prefer to exactly model the means of each random variable but still retain tractability by restricting the greedy learning algorithm to select outer-planar graphs. This model faithfully represents the marginals of each random variable but at the cost of modeling fewer pairwise interactions among the variables.

This is equivalent to the following procedure. First, given the sample moments $\{\mu_i\}$ and $\{\mu_{ij}\}$ we convert these to an equivalent set of zero-mean moments $\widehat{\mu}$ on the extended

(a) 7 × 7 grid (b) Loglikelihood versus number edges (c) Comparison of algorithms

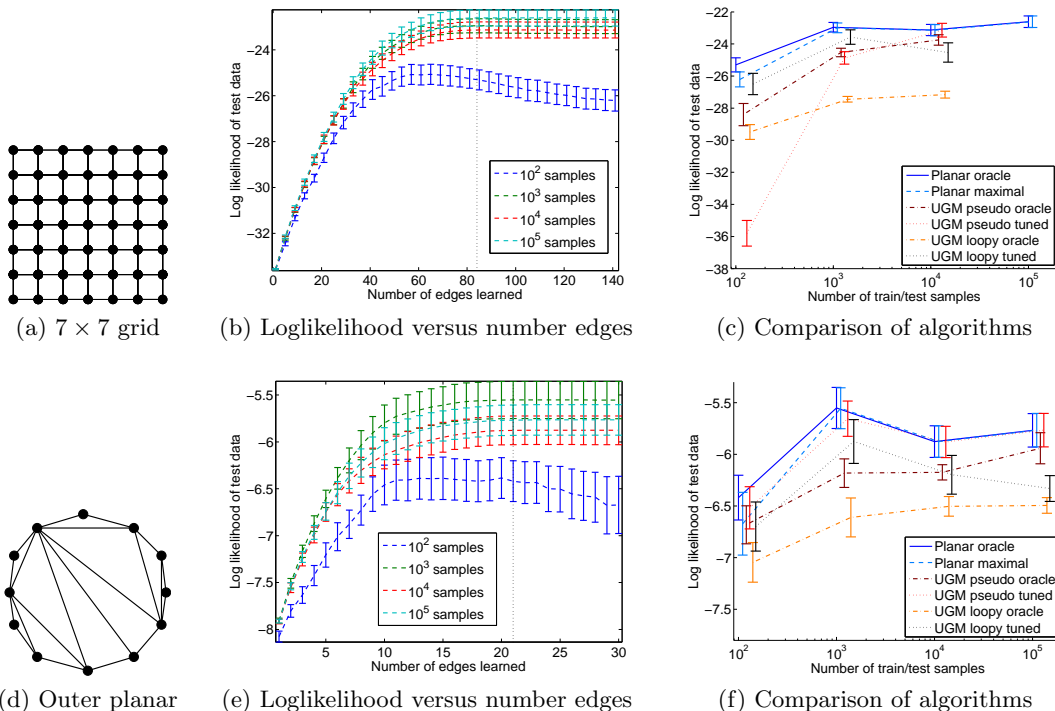(d) Outer planar (e) Loglikelihood versus number edges (f) Comparison of algorithms

Figure 1: Results on known models, (**top** row): 7×7 grid; and (**bottom** row): outer planar. **Left** column (a,d): true graph. **Middle** column (b,e): likelihood of learned planar graphs as edges are added; the true number of edges is marked with a vertical dashed line. **Right** column (c,f): likelihood of test data for various algorithms; x-axis values are perturbed horizontally so that overlapping errorbars are visible.

vertex set $\widehat{V} = V \cup \{n+1\}$ according to Proposition 1. Then, we select a zero-mean planar Ising model for these moments using our greedy algorithm. However, to fit the means of each of the original n variables, we initialize this graph to include all the edges $\{i, n+1\}$ for all $i \in V$ (requiring that these are present in our final estimate of the graph $\widehat{G}$). After this initialization step, we use the same greedy edge-selection procedure as before. This yields the graph $\widehat{G}$ and parameters $\theta_{\widehat{G}}$. Lastly, we convert back to a (non-zero field) Ising model on the subgraph of $\widehat{G}$ defined on nodes V, as prescribed by Proposition 1. The resulting graph G and parameters θ_G is our heuristic solution for the maximum-likelihood outer-planar Ising model.

We remark that it is not essential to choose between the zero-field planar Ising model and the outer-planar Ising model. The greedy algorithm may instead select something in between—a partial outer-planar Ising model where only nodes of the outer-face are allowed to have non-zero means. This is accomplished simply by omitting the initialization step of adding edges $\{i, n+1\}$ for all $i \in V$.

4. Experiments

We present the results of experiments evaluating our algorithm on known models with simulated data to evaluate the correctness of the learned models. We generate two styles of known Ising models: a 7×7 grid ($n = 49$) with zero-field; and a 12-node outer planar model where nodes have non-zero mean; shown in Figures 1a and 1d. The edge parameters are chosen uniformly randomly between -1 and 1 with the condition that the absolute value be greater than a threshold (chosen to be 0.05) so as to avoid edges with negligible interactions. We use Gibbs sampling to obtain samples from this model and calculate empirical moments from these samples which are then passed as input to our algorithm. We run 10 trials of randomly generated edge parameters and data samples. Though our algorithm can run on graphs with many more nodes, we choose small examples here to illustrate the result effectively. On the outer planar model, we ensure that the first moments of all the nodes are satisfied by starting our algorithm with the auxiliary node connected to all other nodes.

As the planar learning algorithm adds edges to the model, the likelihood of the training data is guaranteed to increase. We assess how adding edges affects the likelihood of out-of-sample test data. Figures 1b and 1e demonstrate that likelihood on test sets generally increases as edges are added up to the maximal planar graph. The true number of edges in each synthetic graph is marked with a vertical dotted line. On the smallest data sets (100 samples) the out-of-sample performance begins to degrade, a sign of over-fitting the training data; yet the likelihood of the maximal graph is not significantly worse than the best likelihood obtained (with fewer edges).

We also compare against a Markov random field (MRF) learning algorithm for binary data (Schmidt et al., 2008), as implemented in the undirected graphical model learning Matlab package, UGMLearn[2]. UGM is not restricted to learning planar graphs. The objective is optimized via projected gradient descent. We try two versions of the objective function, one using pseudo-likelihood and the other using loopy belief propagation for inference. UGM employs a regularization parameter which we set using two different methods. First, we used the *tuning* method on validation data as detailed in Schmidt et al. (2008). That is, we split the data into two parts, train on half the data using 7 different values for the parameter, measure the data likelihood of the other half of the data and vice-versa, then select the parameter value that maximizes the validation data likelihood across both folds. The learned model is trained on the full training data with the tuned regularization parameter value. The second method for setting the regularization parameter we call the *oracle* method, where we select the learned model at the true number of edges, k, in our known models. For UGM, we set the regularization parameter via linear search until k edges are learned. The likelihood of test data for the learned UGM model is calculated exactly when $n \leq 25$; in this case, the outer planar example. For larger graphs, e.g. the grid example, the likelihood of test data for UGM is approximated via loopy belief propagation, which we observed to converge well therefore providing a reasonable estimate.

We compare the likelihood of test data from the various learned models in Figures 1c and 1f. For comparison, we selected the maximal planar graph that our algorithm learns, `Planar maximal`; as well as the planar graph learned if the algorithm were stopped when the true number of edges are learned, `Planar oracle`. We compare against `UGM pseudo`

---

2. http://www.cs.ubc.ca/~murphyk/Software/L1CRF

tuned and `UGM loopy tuned`, both of which tune the regularization parameter on validation data; but the former uses pseudo-likelihood in learning and the latter uses loopy belief propagation. The tuning method is the most common way of selecting the regularization parameter, but tends to produce relatively dense graphs. For fair comparison, we also show the likelihood of `UGM pseudo oracle` and `UGM loopy oracle`; that is, the model with the known true number of edges.

Figures 1c and 1f show that our greedy planar Ising model learning algorithm is at least as accurate and often better than the UGM learning algorithms on these inputs. As mentioned earlier, we see that `Planar maximal` and `Planar oracle` fit test data nearly equally well. On the outer planar model, `UGM pseudo tuned` performs nearly as well as our planar algorithm, yet on the larger grid model it performs quite poorly at the smaller sample sizes. `UGM loopy tuned` performs more consistently close to our planar algorithm, but it seems that loopy belief propagation performs worse at large sample sizes.

On the largest data set (10^5 samples) of the 7×7 grid model, UGM was aborted after running for 40 hours without reaching convergence on a single run, and so results are not available.

5. Applications

We apply our planar graph learning algorithm to real-world data in which there is no guarantee that the data is generated according to our model assumptions. The first application models voting patterns of the United States senate, while the second application models geological layers in rocks on Mars. We compare our learned planar graphs against the non-planar graph learning algorithm UGM, as described in the previous section. For the UGM learned graphs, the likelihood of test data is approximated via loopy belief propagation, which we observed to converge well therefore providing a reasonable estimate. Quantitative comparisons indicate that the learned planar models better predict held-out test data with sparser graphs than the non-planar graphs. We also discuss qualitative comparisons of the learned graphs.

5.1 Modeling Correlations of Senator Voting

We consider an interesting application of our algorithm to model correlations of senator voting following Banerjee et al. (2008). We use senator voting data from the years 2009 and 2010 to calculate correlations in the voting patterns among senators. A *Yea* vote is treated as $+1$ and a *Nay* vote is treated as -1. We also treat non-votes as -1, but only consider senators who voted in at least $\frac{3}{4}$ of the votes per year to limit bias. The data includes $n = 108$ variables and 645 samples. To accommodate the non-zero mean data we add an auxiliary node and allow the algorithm to select the connections between it and other nodes. We run a 10-fold cross-validation, training on 90% of the data and measuring likelihood on the held-out 10% of data. Figure 3 shows that the likelihood of test data increases as edges are added. We also show the likelihood of cross-validation test data for the `UGM pseudo` and `UGM loopy` algorithms for two different methods of choosing the value of the regularization parameter: (1) the value that produces the same number of edges as the maximal planar graph (at 318 edges); and (2) the value selected by tuning with validation data (at a variable number of edges, typically a dense graph). The likelihood of the sparse UGM models are
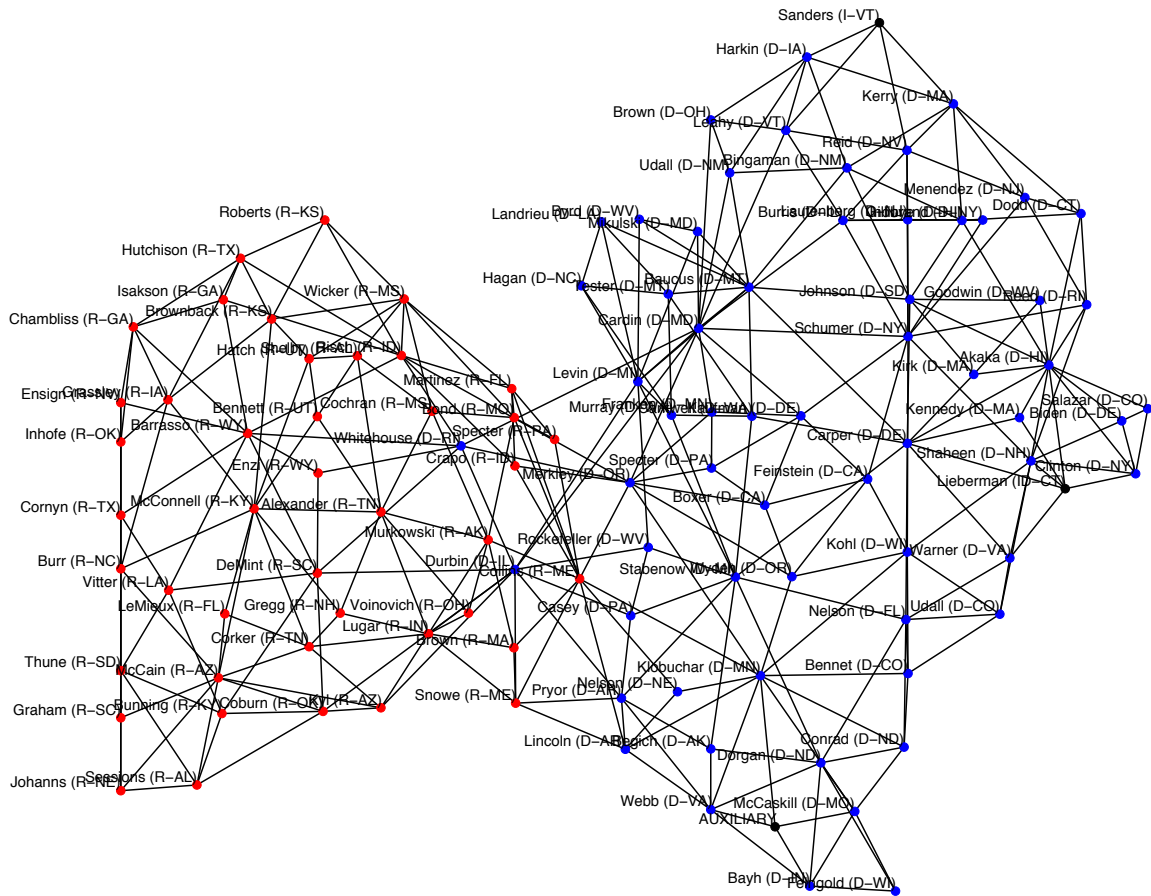
Figure 2: Senator voting results: Learned planar graphical model representing the senator voting pattern. Blue nodes represent Democrats, red nodes represent Republicans and black nodes represents Independents. We use a force-directed graph drawing algorithm (Fruchterman and Reingold, 1991).

significantly worse than the planar model. Only the `UGM loopy` algorithm at a very dense (nearly fully connected) graph has better fit to test data.

The maximal planar graph learned from the full data set, shown in Figure 2, conveys many facts that are already known to us. For instance, the graph shows Sanders with edges only to Democrats which makes sense because he caucuses with Democrats. Same is the case with Lieberman. The graph also shows the senate minority leader McConnell well connected to other Republicans though the same is not true of the senate majority leader Reid. The learned UGM models can be seen in Appendix B, and they show that the non-planar models are qualitatively different, learning one or two densely connected components.
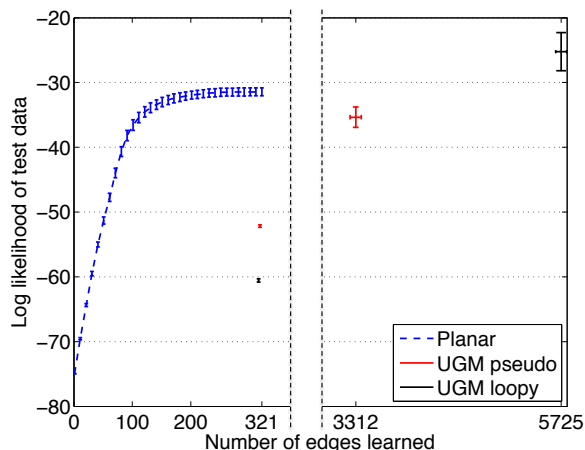
Figure 3: Senator voting results: Comparison of algorithms by likelihood of holdout data versus the number of edges in the learned graph. Note the break in the x-axis, due to tuned UGM learning dense graphs. On the tuned UGM models, we indicate standard error on number of edges learned.
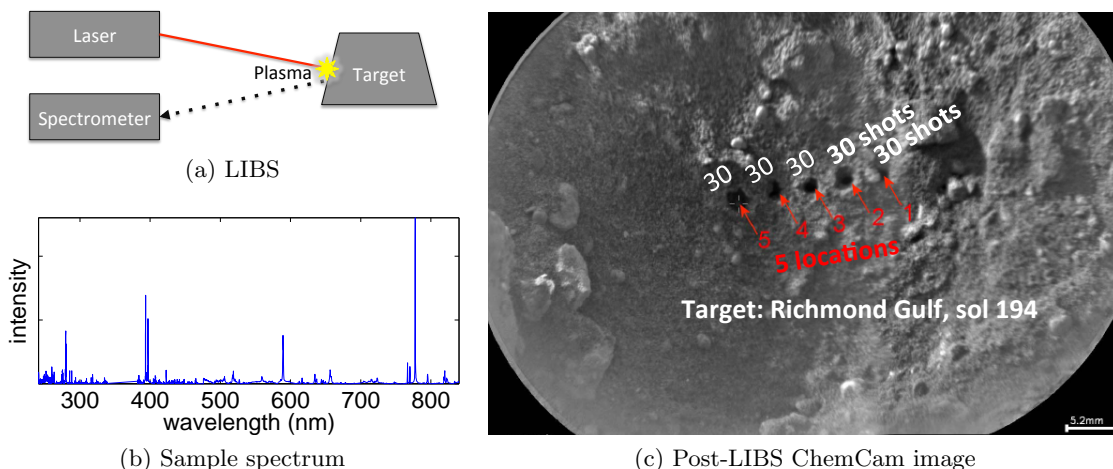

(a) LIBS

(b) Sample spectrum

(c) Post-LIBS ChemCam image

Figure 4: The ChemCam LIBS instrument fires a laser at a target, creating a plasma. Depending on its chemical composition, the plasma gives off different wavelengths of light which are measured by the spectrometer, producing an observed spectrum. Each shot ablates the target surface, leaving a small pit. Typically, sequences of 30 or more shots are fired in several locations on a single target.

5.2 Discovering Depth Trends in Rocks on Mars from Sample Correlations

Our second real-world data set consists of geological observations from the Mars rover *Curiosity*. We are interested in identifying correlations in chemical composition among spatially-related rock samples as taken from the Mars rover *Curiosity*. The ChemCam instrument onboard *Curiosity* collects observations of the chemical composition of rock targets using Laser-Induced Breakdown Spectrometry (LIBS) (Wiens et al., 2012). With each laser

14

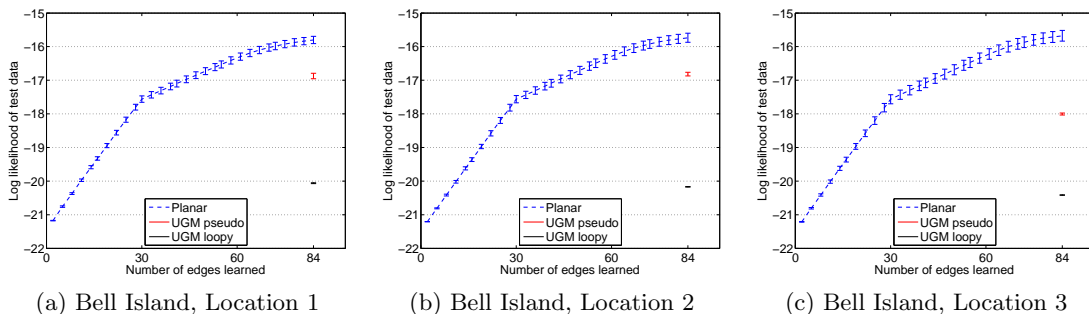(a) Bell Island, Location 1   (b) Bell Island, Location 2   (c) Bell Island, Location 3

Figure 5: Comparison of algorithms on ChemCam data. Likelihood of holdout data versus the number of edges in the learned graph.

shot, the rock surface is ablated and therefore ChemCam produces a sequence of samples at increasing depth; potentially revealing compositional trends such as coatings, weathering rinds and thin stratigraphic layers that could give clues about the past atmospheric and aqueous conditions of Mars (Lanza et al., 2014).

We expect spatial correlations to exist among observations (samples), yet do not expect that the correlations will necessarily correspond to a fixed (known) grid; therefore, our planar model is a reasonable assumption to make. To test this, we compare against the non-planar UGM algorithm.

As shown in Figure 4, each LIBS shot produces a spectral observation consisting of 5810 wavelength bands between 224nm and 840nm. The spectral response is given as a table of intensity values for each wavelength band for each shot. A typical sequence of shots includes 30 - 150 shots on a fixed location. We model the correlations of rock chemistry among these shots, as measured by the set of wavelength bands that show non-zero response (above a noise threshold) in the observed spectra. More precisely, each spectrum is normalized, then thresholded so that 50% of the values are +1. To investigate shot-to-shot correlations, shots are the nodes in the graph while the 5810 wavelength bands are treated as samples. We add an auxiliary node and allow the algorithm to select the connections between it and other nodes. For comparison, we run a 10-fold cross validation using our planar model, the UGM model with pseudolikelihood and loopy belief propagation. Model selection for UGM is done by tuning with validation data and by comparing at the same number of edges learned by the planar algorithm.

We look at 30-shot depth sequences taken at one location at a time to find depth trends of interest. The rock is named Bell Island, and there are three locations that we investigated[3]. The graphs learned by our planar algorithm are quantitatively better than those learned by UGM, as shown in Figure 5. We compare the log-likelihood of 10-fold cross validation data for Planar, UGM pseudo and UGM loopy. With the number of edges in the graph fixed at $3n - 6$ (the number of edges in a maximal planar graph), we see that Planar achieves a significantly better fit to holdout data than UGM with either objective function. When we use the standard tuning method for UGM, it selects dense graphs that

3. NASA data is archived and available at http://pds-geosciences.wustl.edu/missions/msl/chemcam.htm.

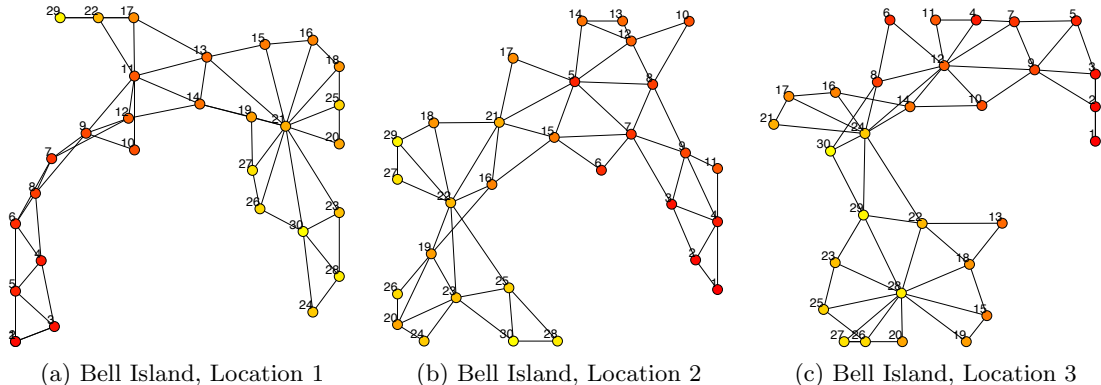(a) Bell Island, Location 1   (b) Bell Island, Location 2   (c) Bell Island, Location 3

Figure 6: Learned planar graphs from ChemCam data. Nodes are numbered by LIBS laser shot number and color-coded by number starting with dark red at shot 1 and fading to light yellow at shot 30. We use a force-directed graph drawing algorithm (Fruchterman and Reingold, 1991).

are nearly fully-connected. While these tuned graphs learned by UGM do fit the data with higher log-likelihood, they provide no insight into the nature of depth trends.

Figure 6 shows the planar graphs learned from the Bell Island ChemCam data. Bell Island, Location 1 shows an interesting pattern of the first 10 or so shots being related to each other in ascending order, while the last 20 shots are more arbitrarily dependent. This pattern is consistent with the observation that the rock is covered in a layer of dust. As the laser ablates through the dust, each shot is conditionally dependent on the next shot. Once the rock itself is being sampled, the composition is more homogenous. At Locations 2 and 3, there is less dust cover, and so there is less of a "tail" formed by the first few shots than seen at Location 1. However, in all of these graphs, we see that the graphs generally link earlier shots with early shots and later shots with late shots, despite not being given this ordering information. This indicates that the chemical composition of the rock is changing with depth.

6. Conclusion and Future Work

We provide a greedy heuristic to obtain the maximum-likelihood planar Ising model approximation to a collection of binary random variables with known pairwise marginals. The algorithm is simple to implement with the help of known methods for tractable exact inference in planar Ising models, efficient methods for planarity testing and embedding of planar graphs. While limiting the search to planar graphs, our learning model provides key advantages over arbitrary (non-planar) graph learning. Namely, the planar graph is sparse without necessitating a regularization term or tuning parameter. Also, the learned planar graph can be used for efficient inference of hidden values in future partially observed data. Further validating this approach, our empirical results on synthetic data and on real data indicate that our planar graph learning finds solutions that it are competitive with arbitrary (non-planar) graph learning in terms of fitting the distribution well.

Directions for further work are suggested by the methods and results of this paper. Firstly, we know that the greedy algorithm is not guaranteed to find the best planar graph. In Appendix C, we provide an enlightening counterexample in which the combination of the planarity restriction and greedy method prevent the correct model from being learned, because a strong indirect correlation exists that masks the correct combination of weaker direct correlations. That counterexample suggests strategies one might consider to further refine the estimate. One strategy would be to allow the greedy algorithm to prune edges which turn out to be less important once later edges are added. It would also be feasible to implement a multi-step greedy look-ahead search technique for selection of which edge to add (or prune) next.

Currently, our framework only allows learning planar graphical models on the set of observed random variables and requires that all variables are observed in each sample. One could imagine extensions of our approach to handle missing samples or to try to identify hidden variables that were not seen in the data. This concept offers another avenue to achieve a better fit to data that is not well-approximated by a planar graph among just the set of observed nodes, but might be well-approximated as the marginal distribution of a planar model with more nodes.

Acknowledgments

Appendix A. Proofs

Proof [Proposition 1] Let the probability distributions corresponding to G and $\widehat{G}$ be P and $\widehat{P}$ respectively and the corresponding expectations be $\mathbb{E}$ and $\widehat{\mathbb{E}}$ respectively. For the partition function, we have that

$$
\begin{aligned}
\widehat{Z} &= \sum_{x_{\widehat{V}}} \exp\left(\sum_{\{i,j\}\in\widehat{E}} \widehat{\theta}_{ij} x_i x_j\right) \\
&= \sum_{x_{\widehat{V}}} \exp\left(x_{n+1}\sum_{i\in V}\theta_i x_i + \sum_{\{i,j\}\in E}\theta_{ij}x_i x_j\right) \\
&= \sum_{x_V} \exp\left(\sum_{i\in V}\theta_i x_i + \sum_{\{i,j\}\in E}\theta_{ij}x_i x_j\right) + \sum_{x_V}\exp\left(-\sum_{i\in V}\theta_i x_i + \sum_{\{i,j\}\in E}\theta_{ij}x_i x_j\right) \\
&= 2\sum_{x_V}\exp\left(\sum_{i\in V}\theta_i x_i + \sum_{\{i,j\}\in E}\theta_{ij}x_i x_j\right) \\
&= 2Z,
\end{aligned}
$$

where the fourth equality follows from the symmetry between -1 and 1 in an Ising model. For the second part, since $\widehat{P}$ is zero-field, we have that

$$\widehat{\mathbb{E}}[x_i] = 0 \ \forall \ i \in \widehat{V}.$$

Now consider any $\{i, j\} \in E$. If x_{n+1} is fixed to a value of 1, then the model is the same as the original one on V and we have

$$\widehat{\mathbb{E}}[x_i x_j \mid x_{n+1} = 1] = \mathbb{E}[x_i x_j] \ \forall \ \{i, j\} \in E.$$

By symmetry (between -1 and 1) in the model, the same is true for $x_{n+1} = -1$ and so

$$\begin{aligned}\widehat{\mathbb{E}}[x_i x_j] &= \widehat{\mathbb{E}}[x_i x_j \mid x_{n+1} = 1]\widehat{P}(x_{n+1} = 1) + \widehat{\mathbb{E}}[x_i x_j \mid x_{n+1} = -1]\widehat{P}(x_{n+1} = -1) \\ &= \mathbb{E}[x_i x_j].\end{aligned}$$

Fixing x_{n+1} to a value of 1, we have

$$\widehat{\mathbb{E}}[x_i \mid x_{n+1} = 1] = \mathbb{E}[x_i] \ \forall \ i \in V,$$

and by symmetry

$$\widehat{\mathbb{E}}[x_i \mid x_{n+1} = -1] = -\mathbb{E}[x_i] \ \forall \ i \in V.$$

Combining the two equations above, we have

$$\begin{aligned}\widehat{\mathbb{E}}[x_i x_{n+1}] &= \widehat{\mathbb{E}}[x_i \mid x_{n+1} = 1]\widehat{P}(x_{n+1} = 1) + \widehat{\mathbb{E}}[-x_i \mid x_{n+1} = -1]\widehat{P}(x_{n+1} = -1) \\ &= \mathbb{E}[x_i].\end{aligned}$$

$$\blacksquare$$

Proof [Proposition 2] From Theorem 1, we see that the log partition function can be written as

$$\Phi(\theta) = n \log 2 + \sum_{\{i,j\} \in E} \log \cosh \theta_{ij} + \frac{1}{2} \log \det(I - AD),$$

where A and D are as given in Theorem 1. For the derivatives, we have

$$\begin{aligned}\frac{\partial \Phi(\theta)}{\partial \theta_{ij}} &= \tanh \theta_{ij} + \frac{1}{2}\mathrm{Tr}\left((I - AD)^{-1}\frac{\partial(I - AD)}{\partial \theta_{ij}}\right) \\ &= \tanh \theta_{ij} - \frac{1}{2}\mathrm{Tr}\left((I - AD)^{-1}AD'_{ij}\right) \\ &= w_{ij} - \frac{1}{2}(1 - w_{ij})^2\left(S_{ij,ij} + S_{ji,ji}\right),\end{aligned}$$

where D'_{ij} is the derivative of the matrix D with respect to θ_{ij}. The first equality follows from the chain rule and the fact that $\nabla \ln |K| = K^{-1}$ for any matrix K. Please refer to Boyd and Vandenberghe (2004) for details.

For the Hessian, we have

$$\begin{aligned}\frac{\partial^2 \Phi(\theta)}{\partial \theta_{ij}^2} &= \frac{1}{Z(\theta)}\frac{\partial^2 Z(\theta)}{\partial \theta_{ij}^2} - \frac{1}{Z(\theta)^2}\left(\frac{\partial Z(\theta)}{\partial \theta_{ij}}\right)^2 \\ &= 1 - \mu_{ij}^2.\end{aligned}$$

For $\{i, j\} \neq \{k, l\}$, following Boyd and Vandenberghe (2004), we have

$$\begin{aligned}\frac{\partial^2 \Phi(\theta)}{\partial \theta_{ij}\partial \theta_{kl}} &= -\frac{1}{2}\mathrm{Tr}\left(SD'_{ij}SD'_{kl}\right) \\ &= -\frac{1}{2}(1 - w_{ij}^2)\left(S_{ij,kl}S_{kl,ij} + S_{ji,kl}S_{kl,ji} + S_{ij,lk}S_{lk,ij} + S_{ji,lk}S_{lk,ji}\right)(1 - w_{kl}^2).\end{aligned}$$

On the other hand, we also have

$$
\begin{aligned}
T_{ij,kl} &= e_{ij}^T(I+P)(S \circ S^T)(I+P)e_{kl} \\
&= (e_{ij}+e_{ji})^T(S \circ S^T)(e_{kl}+e_{lk}) \\
&= (S \circ S^T)_{ij,kl} + (S \circ S^T)_{ij,lk} + (S \circ S^T)_{ji,kl} + (S \circ S^T)_{ji,lk} \\
&= S_{ij,kl}S_{kl,ij} + S_{ji,kl}S_{kl,ji} + S_{ij,lk}S_{lk,ij} + S_{ji,lk}S_{lk,ji},
\end{aligned}
$$

where e_{ij} is the unit vector with 1 in the ij^{th} position and 0 everywhere else. Using the above two equations, we obtain

$$
H_{ij,kl} = -\frac{1}{2}(1-w_{ij}^2)T_{ij,kl}(1-w_{kl}^2).
$$

■

Proof [**Proposition 4**] The proof follows from the following steps of inequalities. The Pythagorean law of information projection (Amari et al., 1992) gives

$$
D(P, P_G) = D(P, P_{G+ij}) + D(P_{G+ij}, P_G).
$$

The conditional rule of relative entropy (Cover and Thomas, 2006) gives

$$
D(P_{G+ij}, P_G) = D(P_{G+ij}(x_i, x_j), P_G(x_i, x_j)) + D(P_{G+ij}(x_{V-ij}|x_i, x_j), P_G(x_{V-ij}|x_i, x_j)),
$$

where $P_{G+ij}(x_i, x_j)$ and $P_G(x_i, x_j)$ represent the marginal distributions on x_i, x_j of probabiliities P_{G+ij} and P_G respectively. Information inequality (Cover and Thomas, 2006) gives us:

$$
D(P_{G+ij}(x_{V-ij}|x_i, x_j), P_G(x_{V-ij}|x_i, x_j)) \geq 0.
$$

Plugging the above two properties into the first equation leads to the inequality

$$
D(P, P_G) \geq D(P, P_{G+ij}) + D(P_{G+ij}(x_i, x_j), P_G(x_i, x_j)).
$$

Finally, the property of information projection to $G+ij$ (Wainwright and Jordan, 2008) gives us $D(P_{G+ij}(x_i, x_j), P_G(x_i, x_j)) \geq D(P(x_i, x_j), P_G(x_i, x_j))$ leading to

$$
D(P, P_G) \geq D(P, P_{G+ij}) + D(P(x_i, x_j), P_G(x_i, x_j)).
$$

■

Appendix B. Applications: UGM Learned Models

For comparison to our planar learning algorithm, we provide the results of using the UGM MRF learning algorithm on the senate voting data and the ChemCam data. For all figures, we use a force-directed graph drawing algorithm (Fruchterman and Reingold, 1991). Figure 7 presents the graph learned using pseudolikelihood, `UGM pseudo`, from the full data set with the regularization parameter set to obtain the same number of edges as learned in the planar case ($3n-6$ edges).
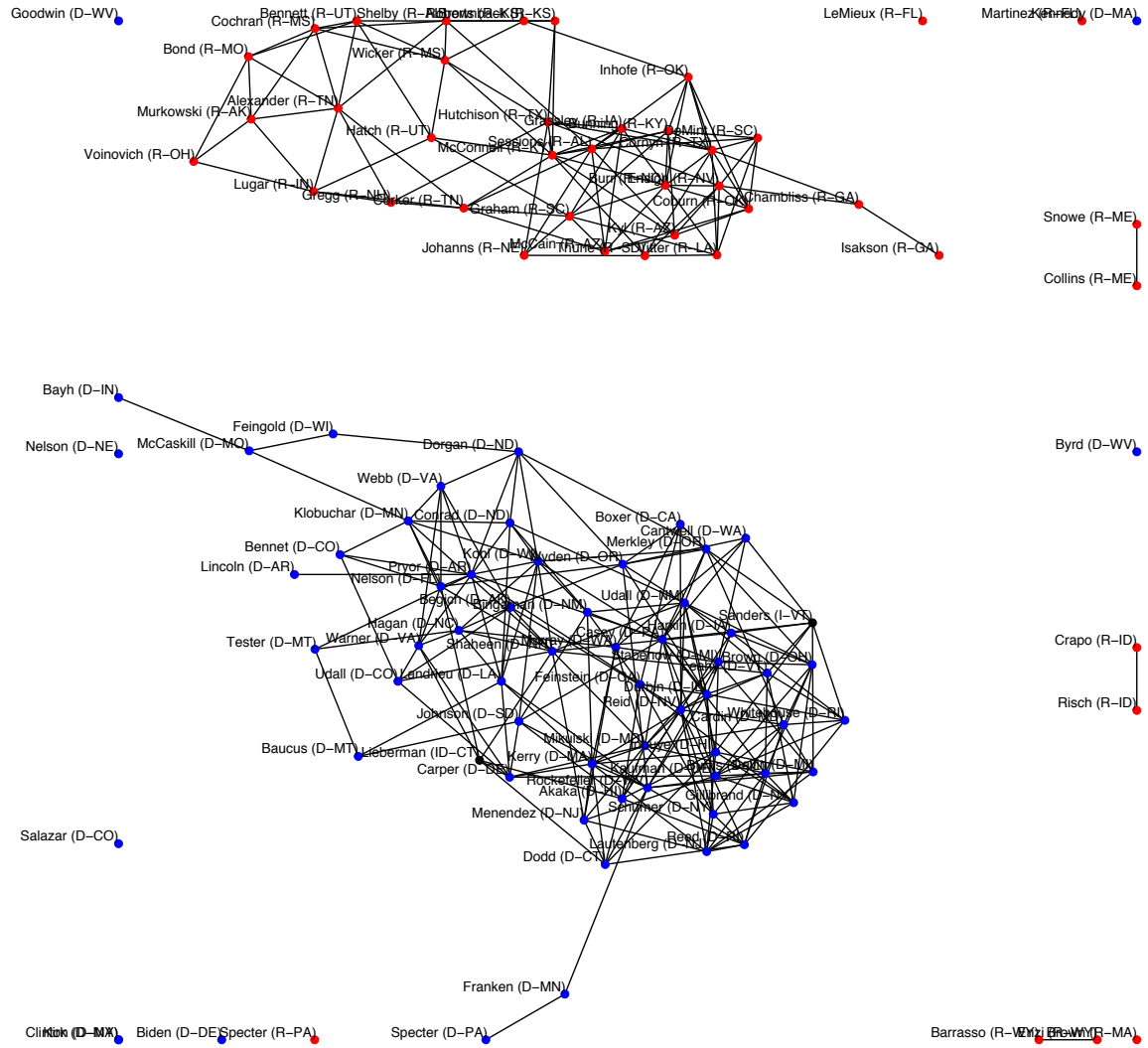
Figure 7: Senate voting graph learned by `UGM pseudo` with 318 edges. Blue nodes represent Democrats, red nodes represent Republicans and black nodes represent Independents.
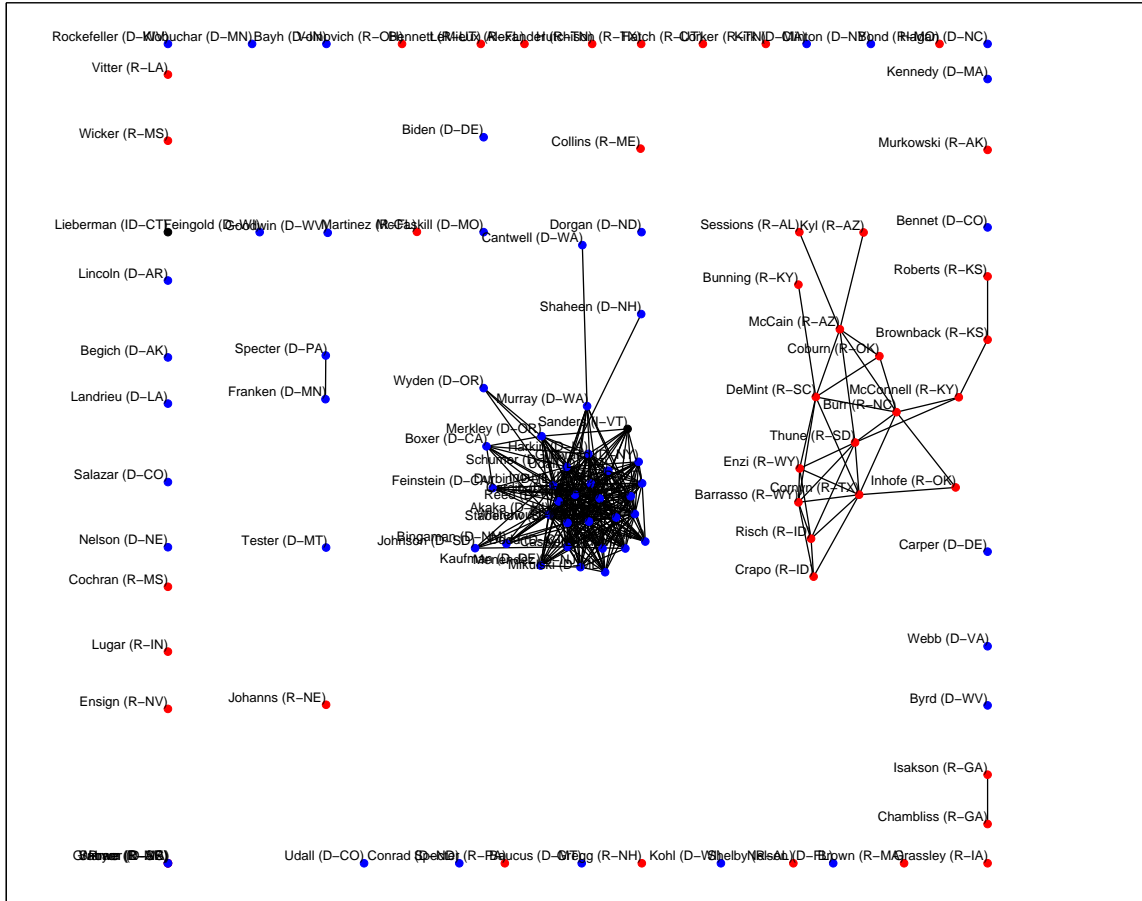
Figure 8: Senate voting graph learned by `UGM loopy` with 318 edges. Blue nodes represent Democrats, red nodes represent Republicans and black nodes represent Independents.

(a) Bell Island, Location 1   (b) Bell Island, Location 2   (c) Bell Island, Location 3
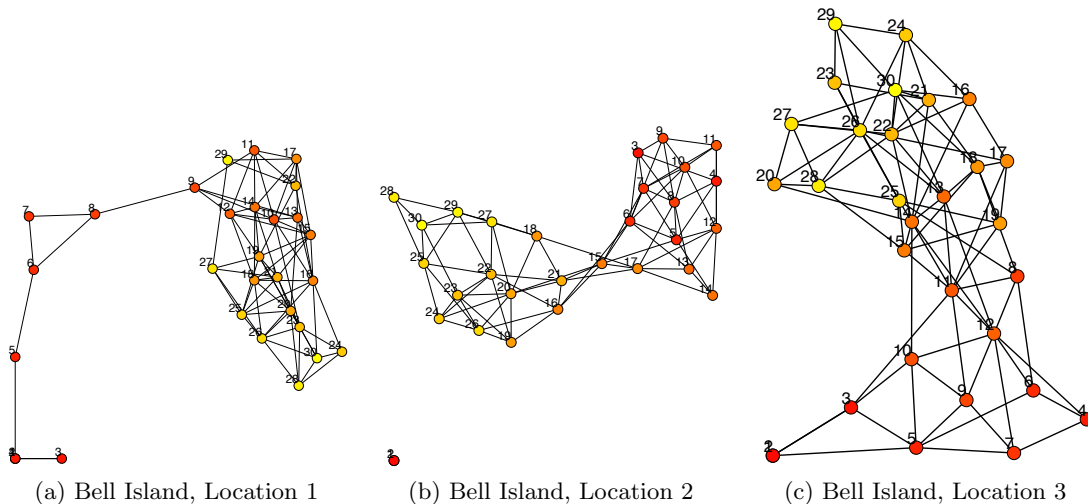
Figure 9: Learned `UGM Pseudo` graphs from ChemCam data. Nodes are numbered by LIBS laser shot number and color-coded by number starting with dark red at shot 1 and fading to light yellow at shot 30. We use a force-directed graph drawing algorithm (Fruchterman and Reingold, 1991).

UGM graphs learned from the senate voting data are given in Figures 7 – 8. Figure 7 presents the graph learned using pseudolikelihood, `UGM pseudo`, from the full data set with the regularization parameter set to obtain 318 edges. Figure 8 presents the graph learned using loopy belief propagation, `UGM loopy`, from the full data set with the regularization parameter set to obtain 318 edges. The graphs learned using the tuning method are not displayed because they are nearly fully-connected graphs providing little visual information.

UGM graphs learned from the ChemCam data from the Bell Island target are given in Figures 9 – 10. Figure 9 presents the graph learned using pseudolikelihood, `UGM pseudo`, from the full data set with the regularization parameter set to obtain 84 edges. Figure 10 presents the graph learning using loopy belief propagation, `UGM loopy`, from the full data set with the regularization parameter set to obtain 84 edges. Graphs learned from this data using the tuning method to select the number of edges are nearly fully connected, and therefore provide little visual information.

Appendix C. Discussion: Counter Example

The result presented in Figure 11 illustrates the fact that our algorithm does not always recover the exact structure even when the underlying graph is planar and the algorithm is given exact moments as inputs. This counterexample gives insight into how the greedy algorithm works. The basic idea is that graphical models can have nodes which are not neighbors but are more correlated than some other nodes which are neighbors. If the spurious edges corresponding to these highly correlated nodes are added early on in the algorithm, then the actual edges may have to be left out because of the planarity restriction.
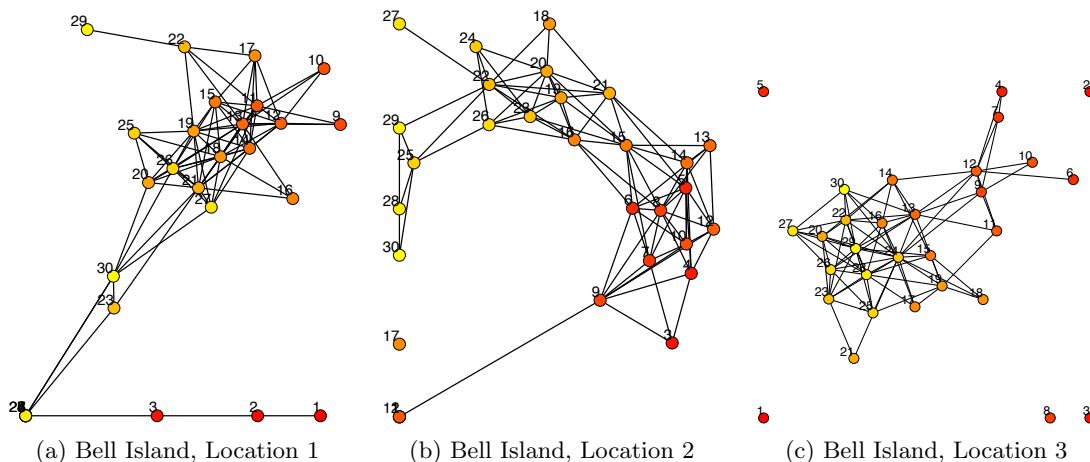
22

(a) Bell Island, Location 1   (b) Bell Island, Location 2   (c) Bell Island, Location 3

Figure 10: Learned `UGM Loopy` graphs from ChemCam data. Nodes are numbered by LIBS laser shot number and color-coded by number starting with dark red at shot 1 and fading to light yellow at shot 30. We use a force-directed graph drawing algorithm (Fruchterman and Reingold, 1991).

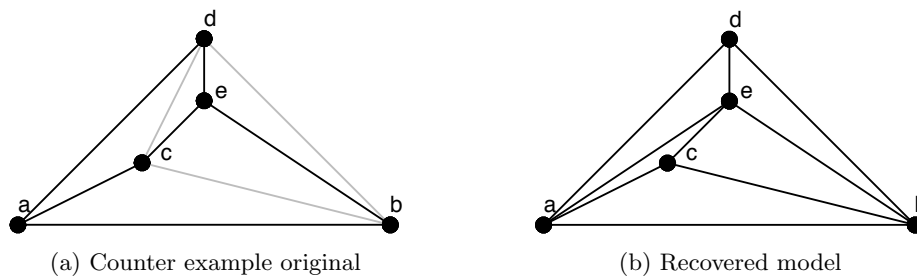(a) Counter example original   (b) Recovered model

Figure 11: Example graphical models. (a) Counter example. (b) The recovered graphical model has one spurious edge $\{a, e\}$ and one missing edge $\{c, d\}$.

We define a zero-field Ising model on the graph in Figure 11a with the edge parameters as follows: $\theta_{bc} = \theta_{cd} = \theta_{bd} = 0.1$ and $\theta_{ij} = 1$ for all the other edges. Figure 11a shows the edge parameters in the graph pictorially using the intensity of the edges - the higher the intensity of an edge, higher the corresponding edge parameter. With these edge parameters, the correlation between nodes a and e is greater than the correlation between any other pair of nodes. This leads to the edge between a and e to be the first edge added in the algorithm. However, since $K5$ (the complete graph on 5 nodes) is not planar, one of the actual edges is missed in the output graph as shown in Figure 11b.

References

P. Abbeel, D. Koller, and A.Y. Ng. Learning factor graphs in polynomial time and sample complexity. *Journal of Machine Learning Research*, 7, 2006.

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 1974.

S. Amari, K. Kurata, and H. Nagaoka. Information geometry of Boltzmann machines. *IEEE Transactions on Neural Networks*, 3(2), 1992.

F. Bach and M. Jordan. Thin junction trees. In *Advances in Neural Information Processing Systems (NIPS)*, 2001.

O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research*, 9, 2008.

A. Barabasi and Z. Oltvai. Network biology: Understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

O. Barndorff-Nielsen. Information and exponential families in statistical theory. *Bulletin of the American Mathematics Society*, 1979.

D. Batra, A. Gallagher, D. Parikh, and T. Chen. Beyond trees: MRF inference via outer-planar decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2496–2503. IEEE, 2010.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14, 1968.

M. Chrobak and T. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Information Processing Letters*, 54(4), 1995.

T. Cover and J. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

A. Deshpande, M. Garofalakis, and M. Jordan. Efficient stepwise selection in decomposable models. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2001.

M. Fisher. On the dimer solution of planar Ising models. *Journal of Mathematical Physics*, 7(10), 1966.

T. Fruchterman and E. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

A. Galluccio, M. Loebl, and J. Vondrak. New algorithm for the Ising problem: Partition function for finite lattice graphs. *Physical Review Letters*, 84(26), 2000.

V. Gómez, H. Kappen, and M. Chertkov. Approximate inference on planar graphs using loop calculus and belief propagation. *The Journal of Machine Learning Research*, 11: 1273–1296, 2010.

A. Jaakkola and Globerson T. Approximate inference using planar graph decomposition. *Advances in Neural Information Processing Systems (NIPS)*, 19:473, 2007.

M. Kac and J. Ward. A combinatorial solution of the two-dimensional Ising model. *Physical Review*, 88(6), 1952.

D. Karger and N. Srebro. Learning Markov networks: Maximum bounded tree-width graphs. In *ACM-SIAM Symposium on Discrete Algorithms*, 2001.

P. Kasteleyn. Dimer statistics and phase transitions. *Journal of Mathematical Physics*, 4 (2), 1963.

N. Lanza, A. Ollila, A. Cousin, R. Wiens, S. Clegg, N. Mangold, N. Bridges, D. Cooper, M. Schmidt, J. Berger, et al. Understanding the signature of rock coatings in laser-induced breakdown spectroscopy data. *Icarus*, 2014.

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

S. Lee, V. Ganapathi, and D. Koller. Efficient structure learning of Markov networks using l1-regularization. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.

R. Lipton and R. Tarjan. A separator theorem for planar graphs. *SIAM Journal of Applied Math*, 36(2), 1979.

R. Lipton, D. Rose, and R. Tarjan. Generalized nested dissection. *SIAM Journal of Numerical Analysis*, 16(2), 1979.

M. Loebl. *Discrete Mathematics in Statistical Physics*. Vieweg + Teubner, 2010.

D. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

L. Onsager. Crystal statistics. I. A Two-dimensional model with an order-disorder transition. *Physical Review*, 65(3-4), 1944.

F. Pozzi, T. Di Matteo, and T. Aste. Spread of risk across financial markets: Better to invest in the peripheries. *Scientific Reports*, 3, 2013.

P. Ravikumar, M. Wainwright, and J. Lafferty. High-dimensional graphical model selection using ℓ_1-regularized logistic regression. *Annals of Statistics*, 38(3), 2010.

D. Richeson. *Euler's Gem: The Polyhedron Formula and the Birth of Topology.* Princeton University Press, 2012.

M. Schmidt, K. Murphy, G. Fung, and R. Rosales. Structure learning in random fields for heart motion abnormality detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

N. Schraudolph and D. Kamenetsky. Efficient exact inference in planar Ising models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1417–1424, 2008.

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2), 1978.

D. Shahaf, A. Checketka, and C. Guestrin. Learning thin junction trees via graph cuts. In *Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009.

S. Sherman. Combinatorial aspects of the Ising model for ferromagnetism. I. A Conjecture of Feynman on paths and graphs. *Journal of Mathematical Physics*, 1(3), 1960.

S. Smith, K. Miller, G. Salimi-Khorshidi, M. Webster, C. Beckmann, T. Nichols, J. Ramsey, and M. Woolrich. Network modelling methods for fMRI. *NeuroImage*, 54(2):875–891, 2011.

M. Wainwright and M. Jordan. *Graphical Models, Exponential Families, and Variational Inference.* Now Publishers Inc., 2008.

R. Wiens, S. Maurice, B. Barraclough, M. Saccoccio, W. Barkley, J. Bell III, S. Bender, J. Bernardin, D. Blaney, J. Blank, et al. The ChemCam instrument suite on the Mars Science Laboratory (MSL) rover: Body unit and combined system tests. *Space Science Reviews*, 170(1-4):167–227, 2012.

J. Yarkony, A. Ihler, and C. Fowlkes. Fast planar correlation clustering for image segmentation. In *European Conference on Computer Vision (ECCV)*, pages 568–581. Springer, 2012.

P. Zhang. Model selection via multifold cross validation. *Annals of Statistics*, 21(1), 1993.