Gaussian Approximation of Collective Graphical Models

Li-Ping Liu[1]   LIULI@EECS.OREGONSTATE.EDU
Daniel Sheldon[2]   SHELDON@CS.UMASS.EDU
Thomas G. Dietterich[1]   TGD@EECS.OREGONSTATE.EDU

[1]School of EECS, Oregon State University, Corvallis, OR 97331 USA
[2]University of Massachusetts, Amherst, MA 01002 and Mount Holyoke College, South Hadley, MA 01075

Abstract

The Collective Graphical Model (CGM) models a population of independent and identically distributed individuals when only collective statistics (i.e., counts of individuals) are observed. Exact inference in CGMs is intractable, and previous work has explored Markov Chain Monte Carlo (MCMC) and MAP approximations for learning and inference. This paper studies Gaussian approximations to the CGM. As the population grows large, we show that the CGM distribution converges to a multivariate Gaussian distribution (GCGM) that maintains the conditional independence properties of the original CGM. If the observations are exact marginals of the CGM or marginals that are corrupted by Gaussian noise, inference in the GCGM approximation can be computed efficiently in closed form. If the observations follow a different noise model (e.g., Poisson), then expectation propagation provides efficient and accurate approximate inference. The accuracy and speed of GCGM inference is compared to the MCMC and MAP methods on a simulated bird migration problem. The GCGM matches or exceeds the accuracy of the MAP method while being significantly faster.

1. Introduction

Consider a setting in which we wish to model the behavior of a population of independent and identically distributed (i.i.d.) individuals but where we can only observe collective count data. For example, we might wish to model the relationship between education, sex, housing, and income from census data. For privacy reasons, the Census Bureau only releases count data such as the number of people hav-

ing a given level of education or the number of men living in a particular region. Another example concerns modeling the behavior of animals from counts of (anonymous) individuals observed at various locations and times. This arises in modeling the migration of fish and birds.

The CGM is constructed by first defining the *individual model*—a graphical model describing a single individual. Let $\mathcal{C}$ and $\mathcal{S}$ be the clique set and the separator set of a junction tree constructed from the individual model. Then, we define N copies of this individual model to create a population of N i.i.d. individuals. This permits us to define count variables $\mathbf{n}_A$, where $\mathbf{n}_A(i_A)$ is the number of individuals for which clique $A \in \mathcal{C} \cup \mathcal{S}$ is in configuration i_A. The counts $\mathbf{n} = (\mathbf{n}_A : A \in \mathcal{C} \cup \mathcal{S})$ are the sufficient statistics of the individual model. After marginalizing away the individuals, the CGM provides a model for the joint distribution of $\mathbf{n}$.

In typical applications of CGMs, we make noisy observations $\mathbf{y}$ that depends on some of the $\mathbf{n}$ variables, and we seek to answer queries about the distribution of some or all of the $\mathbf{n}$ conditioned on these observations. Let $\mathbf{y} = (\mathbf{y}_D : D \in \mathcal{D})$, where $\mathcal{D}$ is a set of cliques from the individual graphical model and $\mathbf{y}_D$ contains counts of settings of clique D. We require each $D \subseteq A$ for some clique $A \in \mathcal{C} \cup \mathcal{S}$ the individual model. In addition to the usual role in graphical models, the inference of the distribution of $\mathbf{n}$ also serves to estimate the parameters of the individual model (e.g. E step in EM learning), because $\mathbf{n}$ are sufficient statistics of the individual model. Inference for CGMs is much more difficult than for the individual model. Unlike the individual model, many conditional distributions in the CGM do not have a closed form. The space of possible configurations of the CGM is very large, because each count variable $\mathbf{n}_i$ can take values in $\{0, \ldots, N\}$.

The original CGM paper, Sheldon and Dietterich (2011) introduced a Gibbs sampling algorithm for sampling from $P(\mathbf{n}|\mathbf{y})$. Subsequent experiments showed that this exhibits slow mixing times, which motivated Sheldon, Sun, Kumar, and Dietterich (2013) to introduce an efficient algorithm

for computing a MAP approximation based on minimizing a tractable convex approximation of the CGM distribution. Although the MAP approximation still scales exponentially in the domain size L of the individual-model variables, it was fast enough to permit fitting CGMs via EM on modest-sized instances ($L = 49$). However, given that we wish to apply this to problems where $L = 1000$, we need a method that is even more efficient.

This paper introduces a Gaussian approximation to the CGM. Because the count variables $\mathbf{n}_C$ have a multinomial distribution, it is reasonable to apply the Gaussian approximation. However, this approach raises three questions. First, is the Gaussian approximation asymptotically correct? Second, can it maintain the sparse dependency structure of the CGM distribution, which is critical to efficient inference? Third, how well does it work with natural (non-Gaussian) observation distributions for counts, such as the Poisson distribution? This paper answers these questions by proving an asymptotically correct Gaussian approximation for CGMs. It shows that this approximation, when done correctly, is able to preserve the dependency structure of the CGM. And it demonstrates that by applying expectation propagation (EP), non-Gaussian observation distributions can be handled. The result is a CGM inference procedure that gives good accuracy and achieves significant speedups over previous methods.

Beyond CGMs, our main result highlights a remarkable property of discrete graphical models: the asymptotic distribution of the vector of sufficient statistics is a Gaussian graphical model with the same conditional independence properties as the original model.

2. Problem Statement and Notation

Consider a graphical model defined on the graph $G = (V, E)$ with n nodes and clique set $\mathcal{C}$. Denote the random variables by $X_1, \ldots, X_n$. Assume for simplicity all variables take values in the same domain $\mathcal{X}$ of size L. Let $\mathbf{x} \in \mathcal{X}^n$ be a particular configuration of the variables, and let $\mathbf{x}_C$ be the subvector of variables belonging to C. For each clique $C \in \mathcal{C}$, let $\phi_C(\mathbf{x}_C)$ be a non-negative potential function. Then the probability model is:

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \phi_C(\mathbf{x}_C)$$
$$= \exp\Big(\sum_{C \in \mathcal{C}} \sum_{i_C \in \mathcal{X}^{|C|}} \theta_C(i_C) \cdot \mathbb{I}(\mathbf{x}_C = i_C) - Q(\boldsymbol{\theta}) \Big). \tag{1}$$

The second line shows the model in exponential-family form Wainwright & Jordan (2008), where $\mathbb{I}(\pi)$ is an indicator variable for the event or expression π, and $\theta_C(i_C) = \log \phi_C(i_C)$ is an entry of the vector of natural parame-

ters. The function $Q(\boldsymbol{\theta}) = \log Z$ is the log-partition function. Given a fixed set of parameters $\boldsymbol{\theta}$ and any subset $A \subseteq V$, the *marginal distribution* $\boldsymbol{\mu}_A$ is the vector with entries $\mu_A(i_A) = \Pr(X_A = i_A)$ for all possible $i_A \in \mathcal{X}^{|A|}$. In particular, we will be interested in the clique marginals $\boldsymbol{\mu}_C$ and the node marginals $\boldsymbol{\mu}_i := \boldsymbol{\mu}_{\{i\}}$.

Junction Trees. Our development relies on the existence of a *junction tree* (Lauritzen, 1996) on the cliques of $\mathcal{C}$ to write the relevant CGM and GCGM distributions in closed form. Henceforth, *we assume that such a junction tree exists*. In practice, this means that one may need to add fill-in edges to the original model to obtain the *triangulated* graph G, of which $\mathcal{C}$ is the set of maximal cliques. This is a clear limitation for graphs with high tree-width. Our methods apply directly to trees and are most practical for low tree-width graphs. Since we use few properties of the junction tree directly, we review only the essential details here and review the reader to Lauritzen (1996) for further details. Let C and C' be two cliques that are adjacent in $\mathcal{T}$; their intersection $S = C \cap C'$ is called a *separator*. Let $\mathcal{S}$ be the set of all separators of $\mathcal{T}$, and let $\nu(S)$ be the number of times S appears as a separator, i.e., the number of different edges (C, C') in $\mathcal{T}$ for which $S = C \cap C'$.

The CGM Distribution. Fix a sample size N and let $\mathbf{x}^1, \ldots, \mathbf{x}^N$ be N i.i.d. random vectors distributed according to the graphical model G. For any set $A \subseteq V$ and particular setting $i_A \in \mathcal{X}^{|A|}$, define the count

$$\mathbf{n}_A(i_A) = \sum_{m=1}^{N} \mathbb{I}(\mathbf{x}_A^m = i_A). \tag{2}$$

Let $\mathbf{n}_A = (\mathbf{n}_A(i_A) : i_A \in \mathcal{X}^{|A|})$ be the complete vector of counts for all possible settings of the variables in A. In particular, let $\mathbf{n}_u := \mathbf{n}_{\{u\}}$ be the vector of node counts. Also, let $\mathbf{n} = (\mathbf{n}_A : A \in \mathcal{C} \cup \mathcal{S})$ be the combined vector of all clique and separator counts—these are sufficient statistics of the sample of size N from the graphical model. The distribution over this vector is the CGM distribution.

Proposition 1 *Let $\mathbf{n}$ be the vector of (clique and separator) sufficient statistics of a sample of size N from the discrete graphical model* (1). *The probability mass function of $\mathbf{n}$ is given by $p(\mathbf{n}; \boldsymbol{\theta}) = h(\mathbf{n})f(\mathbf{n}; \boldsymbol{\theta})$ where*

$$f(\mathbf{n}; \boldsymbol{\theta}) = \exp\Big(\sum_{C \in \mathcal{C}, i_C \in \mathcal{X}^{|C|}} \theta_C(i_C) \cdot \mathbf{n}_C(i_C) - NQ(\boldsymbol{\theta}) \Big) \tag{3}$$

$$h(\mathbf{n}) = N! \cdot \frac{\prod_{S \in \mathcal{S}} \prod_{i_S \in \mathcal{X}^{|S|}} \left(\mathbf{n}_S(i_S)!\right)^{\nu(S)}}{\prod_{C \in \mathcal{C}} \prod_{i_C \in \mathcal{X}^{|C|}} \mathbf{n}_C(i_C)!}$$

$$\prod_{S \sim C \in \mathcal{T}, i_S \in \mathcal{X}^{|S|}} \mathbb{I}\left(\mathbf{n}_S(i_S) = \sum_{i_{C \setminus S}} \mathbf{n}_C(i_S, i_{C \setminus S})\right) \cdot$$

$$\prod_{C \in \mathcal{C}} \mathbb{I}\left(\sum_{i_C \in \mathcal{X}^{|C|}} \mathbf{n}_C(i_C) = N\right). \quad (4)$$

Denote this distribution by $\mathrm{CGM}(N, \boldsymbol{\theta})$.

Here, the notation $S \sim C \in \mathcal{T}$ means that S is adjacent to C in $\mathcal{T}$. This proposition was first proved in nearly this form by Sundberg (1975) (see also Lauritzen (1996)). Proposition 1 differs from those presentations by writing $f(\mathbf{n}; \boldsymbol{\theta})$ in terms of the original parameters $\boldsymbol{\theta}$ instead of the clique and separator marginals $\{\boldsymbol{\mu}_C, \boldsymbol{\mu}_S\}$, and by including hard constraints in the base measure $h(\mathbf{n})$. The hard constraints enforce consistency of the sufficient statistics of all cliques on their adjacent separators, and were treated implicitly prior to Sheldon & Dietterich (2011). A proof of the equivalence between our expression for $f(\mathbf{n}; \boldsymbol{\theta})$ and the expressions from prior work is given in the supplementary material. Dawid & Lauritzen (1993) refer to the same distribution as the *hyper-multinomial* distribution due to the fact that it follows conditional independence properties analogous to those in the original graphical model.

Proposition 2 *Let* $A, B \in \mathcal{S} \cup \mathcal{C}$ *be two sets that are separated by the separator* S *in* $\mathcal{T}$. *Then* $\mathbf{n}_A \perp\!\!\!\perp \mathbf{n}_B \mid \mathbf{n}_S$.

Proof: The probability model $p(\mathbf{n}; \boldsymbol{\theta})$ factors over the clique and separator count vectors $\mathbf{n}_C$ and $\mathbf{n}_S$. The only factors where two different count vectors appear together are the consistency constraints where $\mathbf{n}_S$ and $\mathbf{n}_C$ appear together if S is adjacent to C in $\mathcal{T}$. Thus, the CGM is a graphical model with the same structure as $\mathcal{T}$, from which the claim follows. $\square$

3. Approximating CGM by the Normal Distribution

In this section, we will develop a Gaussian approximation, GCGM, of the CGM and show that it is the asymptotically correct distribution as M goes to infinity. We then show that the GCGM has the same conditional independence structure as the CGM, and we explicitly derive the conditional distributions. These allow us to use Gaussian message passing in the GCGM as a practical approximate inference method for CGMs.

We will follow the most natural approach of approximating the CGM distribution by a multivariate Gaussian with the same mean and covariance matrix. The moments of

the CGM distribution follow directly from those of the indicator variables of the individual model: Fix an outcome $\mathbf{x} = (x_1, \ldots, x_n)$ from the individual model and for any set $A \subseteq V$ let $\mathbf{I}_A = \left(\mathbb{I}(\mathbf{x}_A = i_A) : i_A \in \mathcal{X}^{|A|}\right)$ be the vector of all indicator variables for that set. The mean and covariance of any such vectors are given by

$$\mathbb{E}[\mathbf{I}_A] = \boldsymbol{\mu}_A \quad (5)$$

$$\mathrm{cov}(\mathbf{I}_A, \mathbf{I}_B) = \langle \boldsymbol{\mu}_{A,B} \rangle - \boldsymbol{\mu}_A \boldsymbol{\mu}_B^T. \quad (6)$$

Here, the notation $\langle \boldsymbol{\mu}_{A,B} \rangle$ refers to the matrix whose (i_A, i_B) entry is the marginal probability $\mathrm{Pr}(X_A = i_A, X_B = i_B)$. Note that Eq. (6) follows immediately from the definition of covariance for indicator variables, which is easily seen in the scalar form: $\mathrm{cov}(\mathbb{I}(X_A = i_A), \mathbb{I}(X_B = i_B)) = \mathrm{Pr}(X_A = i_A, X_B = i_B) - \mathrm{Pr}(X_A = i_A)\mathrm{Pr}(X_B = i_B)$. Eq. (6) also covers the case when $A \cap B$ is nonempty. In particular if $A = B = \{u\}$, then we recover $\mathrm{cov}(\mathbf{I}_u, \mathbf{I}_u) = \mathrm{diag}(\boldsymbol{\mu}_u) - \boldsymbol{\mu}_u \boldsymbol{\mu}_u^T$, which is the covariance matrix for the marginal multinomial distribution of $\mathbf{I}_u$.

From the preceding arguments, it becomes clear that the covariance matrix for the full vector of indicator variables has a simple block structure. Define $\mathbf{I} = (\mathbf{I}_A : A \in \mathcal{C} \cup \mathcal{S})$ to be the vector concatention of all the clique and separator indicator variables, and let $\boldsymbol{\mu} = (\boldsymbol{\mu}_A : A \in \mathcal{C} \cup \mathcal{S}) = \mathbb{E}[\mathbf{I}]$ be the corresponding vector concatenation of marginals. Then it follows from (6) that the covariance matrix is

$$\Sigma := \mathrm{cov}(\mathbf{I}, \mathbf{I}) = \hat{\Sigma} - \boldsymbol{\mu}\boldsymbol{\mu}^T, \quad (7)$$

where $\hat{\Sigma}$ is the matrix whose (A, B) block is the marginal distribution $\langle \boldsymbol{\mu}_{A,B} \rangle$. In the CGM model, the count vector $\mathbf{n}$ can be written as $\mathbf{n} = \sum_{m=1}^{N} \mathbf{I}^m$, where $\mathbf{I}^1, \ldots, \mathbf{I}^N$ are i.i.d. copies of $\mathbf{I}$. As a result, the moments of the CGM are obtained by scaling the moments of $\mathbf{I}$ by N. We thus arrive at the natural moment-matching Gaussian approximation of the CGM.

Definition 1 *The Gaussian CGM, denoted* $\mathrm{GCGM}(N, \boldsymbol{\theta})$ *is the multivariate normal distribution* $\mathcal{N}(N\boldsymbol{\mu}, N\Sigma)$, *where* $\boldsymbol{\mu}$ *is the vector of all clique and separator marginals of the graphical model with parameters* $\boldsymbol{\theta}$, *and* Σ *is defined in Equation* (7).

In the following theorem, we show the GCGM is asymptotically correct and it is a Gaussian graphical model, which will lead to efficient inference algorithms.

Theorem 1 *Let* $\mathbf{n}^N \sim \mathrm{CGM}(N, \boldsymbol{\theta})$ *for* $N = 1, 2, \ldots$. *Then following are true:*

(i) *The GCGM is asymptotically correct. That is, as* $N \to \infty$ *we have*

$$\frac{1}{\sqrt{N}}(\mathbf{n}^N - N\boldsymbol{\mu}) \xrightarrow{D} \mathcal{N}(\mathbf{0}, \Sigma). \quad (8)$$

(ii) The GCGM is a Gaussian graphical model with the same conditional independence structure as the CGM. Let $\mathbf{z} \sim \text{GCGM}(N, \boldsymbol{\theta})$ and let $A, B \in \mathcal{C} \cup \mathcal{S}$ be two sets that are separated by separator S in $\mathcal{T}$. Then $\mathbf{z}_A \perp\!\!\!\perp \mathbf{z}_B \mid \mathbf{z}_S$.

Proof: Part (i) is a direct application of the multivariate central limit theorem to the random vector $\mathbf{n}^N$, which, as noted above, is a sum of i.i.d. random vectors $\mathbf{I}^1, \ldots, \mathbf{I}^N$ with mean $\boldsymbol{\mu}$ and covariance Σ (Feller, 1968).

Part (ii) is a consequence of the fact that these conditional independence properties hold for each $\mathbf{n}^N$ (Proposition 2), so they also hold in the limit as $N \to \infty$. While this is intuitively clear, it seems to require further justification, which is provided in the supplementary material. $\square$

3.1. Conditional Distributions

The goal is to use inference in the GCGM as a tractable approximate alternative inference method for CGMs. However, it is very difficult to compute the covariance matrix Σ over all cliques. In particular, note that the (C, C') block requires the joint marginal $\langle \boldsymbol{\mu}_{C,C'} \rangle$, and if C and C' are not adjacent in $\mathcal{T}$ this is hard to compute. Fortunately, we can sidestep the problem completely by leveraging the graph structure from Part (ii) of Theorem 1 to write the distribution as a product of conditional distributions whose parameters are easy to compute (this effectively means working with the inverse covariance matrix instead of Σ). We then perform inference by Gaussian message passing on the resulting model.

A challenge is that Σ is not full rank, so the GCGM distribution as written is degenerate and does not have a density. This can be seen by noting that any vector $\mathbf{n} \sim \text{CGM}(N; \boldsymbol{\theta})$ with nonzero probability satisfies the affine consistency constraints from Eq. (4)—for example, each vector $\mathbf{n}_C$ and $\mathbf{n}_S$ sums to the population size N—and that these affine constraints also hold with probability one in the limiting distribution. To fix this, we instead use a linear transformation $\mathbb{T}$ to map $\mathbf{z}$ to a reduced vector $\tilde{\mathbf{z}} = \mathbb{T}\,\mathbf{z}$ such that the reduced covariance matrix $\tilde{\Sigma} = \mathbb{T}\,\Sigma\,\mathbb{T}^T$ is invertible. The work by Loh & Wainwright (2013) proposed a minimal representation of the graphical model in (1), and the corresponding random variable has a full rank covariance matrix. We will find a transformation $\mathbb{T}$ to project our indicator variable $\mathbf{I}$ into that form. Then $\mathbb{T}\,\mathbf{I}$ (as well as $\mathbb{T}\,\mathbf{n}$ and $\mathbb{T}\,\mathbf{z}$) will have a full rank covariance matrix.

Denote by $\mathcal{C}^+$ the maximal and non-maximal cliques in the triangulated graph. Note that each $D \in \mathcal{C}^+$ must be a subset of some $A \in \mathcal{C} \cup \mathcal{S}$ and each subset of A is also a clique in $\mathcal{C}^+$. For every $D \in \mathcal{C}^+$, let $\mathcal{X}_0^D = (\mathcal{X} \backslash \{L\})^{|D|}$ denote the space of possible configurations of D after excluding the largest value, L, from the domain of each variable in

D. The corresponding random variable $\tilde{\mathbb{I}}$ in the minimal representation is defined as (Loh & Wainwright, 2013):

$$\tilde{\mathbf{I}} = (\mathbb{I}(\mathbf{x}_D = i_D) : i_D \in \mathcal{X}_0^D, D \in \mathcal{C}^+) \; . \tag{9}$$

$\tilde{\mathbf{I}}_D$ can be calculated linearly from $\mathbf{I}_A$ when $D \subseteq A$ via the matrix $\mathbb{T}_{D,A}$ whose (i_D, i_A) entry is defined as

$$\mathbb{T}_{D,A}(i_D, i_A) = \mathbb{I}(i_D \sim_D i_A), \tag{10}$$

where $\sim_D$ means that i_D and i_A agree on the setting of the variables in D. It follows that $\tilde{\mathbf{I}}_D = \mathbb{T}_{D,A}\,\mathbf{I}_A$. The whole transformation $\mathbb{T}$ can be built in blocks as follows: For every $D \in \mathcal{C}^+$, choose $A \in \mathcal{C} \cup \mathcal{S}$ and construct the $\mathbb{T}_{D,A}$ block via (10). Set all other blocks to zero. Due to the redundancy of $\mathbf{I}$, there might be many ways of choosing A for D and any one will work as long as $D \subseteq A$.

Proposition 3 *Define $\mathbb{T}$ as above, and define $\tilde{\mathbf{z}} = \mathbb{T}\,\mathbf{z}$, $\tilde{\mathbf{z}}_{A+} = (\tilde{\mathbf{z}}_D : D \subseteq A), A \in \mathcal{C} \cup \mathcal{S}$. Then*

(i) If $A, B \in \mathcal{C} \cup S$ are separated by S in $\mathcal{T}$, it holds that $\tilde{\mathbf{z}}_{A+} \perp\!\!\!\perp \tilde{\mathbf{z}}_{B+} \mid \tilde{\mathbf{z}}_{S+}$.

(ii) The covariance matrix of $\tilde{\mathbf{z}}$ has full rank.

Proof: In the appendix, we show that for any $A \in \mathcal{C} \cup \mathcal{S}$, $\mathbf{I}_A$ can be linearly recovered from $\tilde{\mathbf{I}}_{A+} = (\tilde{\mathbf{I}}_D : D \subseteq A)$. So there is a linear bijection between $\mathbf{I}_A$ and $\tilde{\mathbf{I}}_{A+}$ (The mapping from $\mathbf{I}_A$ to $\tilde{\mathbf{I}}_{A+}$ has been shown in the definition of $\mathbb{T}$). The same linear bijection relation also exists between $\mathbf{n}_A$ and $\tilde{\mathbf{n}}_{A+} = \sum_{m=1}^N \tilde{\mathbf{I}}_{A+}^m$ and between $\mathbf{z}_A$ and $\tilde{\mathbf{z}}_{A+}$.

Proof of (i): Since $\mathbf{z}_A \perp\!\!\!\perp \mathbf{z}_B \mid \mathbf{z}_S$, it follows that $\tilde{\mathbf{z}}_{A+} \perp\!\!\!\perp \tilde{\mathbf{z}}_{B+} \mid \mathbf{z}_S$ because $\tilde{\mathbf{z}}_{A+}$ and $\tilde{\mathbf{z}}_{B+}$ are deterministic functions of $\mathbf{z}_A$ and $\mathbf{z}_B$ respectively. Since $\mathbf{z}_S$ is a deterministic function of $\tilde{\mathbf{z}}_{S+}$, the same property holds when we condition on $\tilde{\mathbf{z}}_{S+}$ instead of $\mathbf{z}_S$.

Proof of (ii): The bijection between $\mathbf{I}$ and $\tilde{\mathbf{I}}$ indicates that the model representation of Loh & Wainwright (2013) defines the same model as (1). By Loh & Wainwright (2013), $\tilde{\mathbf{I}}$ has full rank covariance matrix and so do $\tilde{\mathbf{n}}$ and $\tilde{\mathbf{z}}$. $\square$

With this result, the GCGM can be decomposed into conditional distributions, and each distribution is a non-degenerate Gaussian distribution.

Now let us consider the observations $\mathbf{y} = \{\mathbf{y}_D, D \in \mathcal{D}\}$, where $\mathcal{D}$ is the set of cliques for which we have observations. We require each $D \in \mathcal{D}$ be subset of some clique $C \in \mathcal{C}$. When choosing a distribution $p(\mathbf{y}_D | \mathbf{z}_C)$, a modeler has substantial flexibility. For example, $p(\mathbf{y}_D | \mathbf{z}_C)$ can be noiseless, $\mathbf{y}_D(i_D) = \sum_{i_{C \backslash D}} \mathbf{z}_C(i_D, i_{C \backslash D})$, which permits closed-form inference. Or $p(\mathbf{y}_D | \mathbf{z}_C)$ can consist of independent noisy observations: $p(\mathbf{y}_D | \mathbf{z}_C) = \prod_{i_D} p(\mathbf{y}_D(i_D) | \sum_{i_{C \backslash D}} \mathbf{z}_C(i_D, i_{C \backslash D}))$. With a little work, $p(\mathbf{y}_D | \mathbf{z}_C)$ can be represented by $p(\mathbf{y}_D | \tilde{\mathbf{z}}_{C+})$.

3.2. Explicit Factored Density for Trees

We describe how to decompose GCGM for the special case when the original graphical model G is a tree. We assume that only counts of single nodes are observed. In this case, we can marginalize out edge (clique) counts $\mathbf{z}_{\{u,v\}}$ and retain only node (separator) counts $\mathbf{z}_u$. Because the GCGM has a normal distribution, marginalization is easy. The conditional distribution is then defined only on node counts. With the definition of $\tilde{\mathbf{z}}$ in Proposition (3) and the property of conditional independence, we can write

$$p(\tilde{\mathbf{z}}_1, \ldots, \tilde{\mathbf{z}}_n) = p(\tilde{\mathbf{z}}_r) \prod_{(u,v) \in E} p(\tilde{\mathbf{z}}_v \mid \tilde{\mathbf{z}}_u). \quad (11)$$

Here $r \in V$ is an arbitrarily-chosen root node, and E is the set of *directed* edges of G oriented away from r. The marginalization of the edges greatly reduces the size of the inference problem, and a similar technique is also applicable to general GCGMs.

Now specify the parameters of the Gaussian conditional densities $p(\tilde{\mathbf{z}}_v \mid \tilde{\mathbf{z}}_u)$ in Eq. (11). Assume the blocks $\mathbb{T}_{u,u}$ and $\mathbb{T}_{v,v}$ are defined as (10). Let $\tilde{\boldsymbol{\mu}}_u = \mathbb{T}_{u,u}\, \boldsymbol{\mu}_u$ be the marginal vector of node u without its last entry, and let $\langle \tilde{\boldsymbol{\mu}}_{u,v} \rangle = \mathbb{T}_{u,u} \langle \boldsymbol{\mu}_{u,v} \rangle \mathbb{T}_{v,v}^T$ be the marginal matrix over edge (u, v), minus the final row and column. Then the mean and covariance martix of the joint distribution are

$$\boldsymbol{\eta} := N \begin{bmatrix} \tilde{\boldsymbol{\mu}}_u \\ \tilde{\boldsymbol{\mu}}_v \end{bmatrix}, \quad N^2 \begin{bmatrix} \mathrm{diag}(\tilde{\boldsymbol{\mu}}_u) & \langle \tilde{\boldsymbol{\mu}}_{u,v} \rangle \\ \langle \tilde{\boldsymbol{\mu}}_{v,u} \rangle & \mathrm{diag}(\tilde{\boldsymbol{\mu}}_v) \end{bmatrix} - \boldsymbol{\eta}\boldsymbol{\eta}^T. \quad (12)$$

The conditional density $p(\tilde{\mathbf{z}}_v \mid \tilde{\mathbf{z}}_u)$ is obtained by standard Gaussian conditioning formulas.

If we need to infer $\mathbf{z}_{\{u,v\}}$ from some distribution $q(\tilde{\mathbf{z}}_u, \tilde{\mathbf{z}}_v)$, we first calculate the distribution $p(\tilde{\mathbf{z}}_{\{u,v\}}|\tilde{\mathbf{z}}_u, \tilde{\mathbf{z}}_v)$. This time we assume blocks $\mathbb{T}_{\{u,v\}^+,\{u,v\}} = (\mathbb{T}_{u,\{u,v\}} : D \in \{u,v\})$ are defined as (10). We can find the mean and variance of $p(\tilde{\mathbf{z}}_u, \tilde{\mathbf{z}}_v, \tilde{\mathbf{z}}_{\{u,v\}})$ by applying linear transformation $\mathbb{T}_{\{u,v\}^+,\{u,v\}}$ on the mean and variance of $\mathbf{z}_{\{u,v\}}$. Standard Gaussian conditioning formulas then give the conditional distribution $p(\tilde{\mathbf{z}}_{\{u,v\}} \mid \tilde{\mathbf{z}}_u, \tilde{\mathbf{z}}_v)$. Then we can recover the distribution of $\mathbf{z}_{\{u,v\}}$ from distribution $p(\tilde{\mathbf{z}}_{\{u,v\}}|\tilde{\mathbf{z}}_u, \tilde{\mathbf{z}}_v)q(\tilde{\mathbf{z}}_u, \tilde{\mathbf{z}}_v)$.

Remark: Our reasoning gives a completely different way of deriving some of the results of Loh & Wainwright (2013) concerning the sparsity pattern of the inverse covariance matrix of the sufficient statistics of a discrete graphical model. The conditional independence in Proposition 2 for the factored GCGM density translates directly to the sparsity pattern in the precision matrix $\Gamma = \tilde{\Sigma}^{-1}$. Unlike the reasoning of Loh & Wainwright, we derive the sparsity directly from the conditional independence properties of the asymptotic distribution (which are inherited from the CGM distribution) and the fact that the CGM and GCGM share the same covariance matrix.

4. Inference with Noisy Observations

We now consider the problem of inference in the GCGM when the observations are noisy. Throughout the remainder of the paper, we assume that the individual model—and, hence, the CGM—is a tree. In this case, the cliques correspond to edges and the separators correspond to nodes. We will also assume that only the nodes are observed. For notational simplicity, we will assume that every node is observed (with noise). (It is easy to marginalize out unobserved nodes if any.) From now on, we use uv instead of $\{u, v\}$ to represent edge clique. Finally, we assume that the entries have been dropped from the vector $\mathbf{z}$ as described in the previous section so that it has the factored density described in Eq. 11.

Denote the observation variable for node u by $\mathbf{y}_u$, and assume that it has a Poisson distribution. In the (exact) CGM, this would be written as $\mathbf{y}_u \sim \mathrm{Poisson}(\mathbf{n}_u)$. However, in our GCGM, this instead has the form

$$\mathbf{y}_u \sim \mathrm{Poisson}(\lambda \mathbf{z}_u), \quad (13)$$

where $\mathbf{z}_u$ is the corresponding continuous variable and λ determines the amount of noise in the distribution. Denote the vector of all observations by $\mathbf{y}$. Note that the missing entry of $\mathbf{z}_u$ must be reconstructed from the remaining entries when computing the likelihood.

With Poisson observations, there is no longer a closed-form solution to message passing in the GCGM. We address this by applying Expectation Propagation (EP) with the Laplace approximation. This method has been previously applied to nonlinear dynamical systems by Ypma and Heskes (2005).

4.1. Inferring Node Counts

In the GCGM with observations, the potential on each edge $(u, v) \in E$ is defined as

$$\psi(\mathbf{z}_u, \mathbf{z}_v) = \begin{cases} p(\mathbf{z}_v, \mathbf{z}_u)p(\mathbf{y}_v|\mathbf{z}_v)p(\mathbf{y}_u|\mathbf{z}_u) & \text{if } u \text{ is root} \\ p(\mathbf{z}_v|\mathbf{z}_u)p(\mathbf{y}_v|\mathbf{z}_v) & \text{otherwise.} \end{cases} \quad (14)$$

We omit the subscripts on ψ for notational simplicity. The joint distribution of $(\mathbf{z}_v, \mathbf{z}_u)$ has mean and covariance shown in (12).

With EP, the model approximates potential on edge $(u, v) \in E$ with normal distribution in context $q_{\backslash uv}(\mathbf{z}_u)$ and $q_{\backslash uv}(\mathbf{z}_v)$. The context for edge (u, v) is defined as

$$q_{\backslash uv}(\mathbf{z}_u) = \prod_{(u,v') \in E, v' \neq v} q_{uv'}(\mathbf{z}_u) \quad (15)$$

$$q_{\backslash uv}(\mathbf{z}_v) = \prod_{(u',v) \in E, u' \neq u} q_{u'v}(\mathbf{z}_v), \quad (16)$$

where each $q_{uv'}(\mathbf{z}_u)$ and $q_{u'v}(\mathbf{z}_v)$ have the form of normal densities.

Let $\xi(\mathbf{z}_u, \mathbf{z}_v) = q_{\setminus uv}(\mathbf{z}_u) q_{\setminus uv}(\mathbf{z}_v) \psi(\mathbf{z}_u, \mathbf{z}_v)$. The EP update of $q_{uv}(\mathbf{z}_u)$ and $q_{uv}(\mathbf{z}_v)$ is computed as

$$q_{uv}(\mathbf{z}_u) = \frac{\text{proj}_{\mathbf{z}_u}[\xi(\mathbf{z}_u, \mathbf{z}_v)]}{q_{\setminus uv}(\mathbf{z}_u)} \qquad (17)$$

$$q_{uv}(\mathbf{z}_v) = \frac{\text{proj}_{\mathbf{z}_v}[\xi(\mathbf{z}_u, \mathbf{z}_v)]}{q_{\setminus uv}(\mathbf{z}_v)}. \qquad (18)$$

The projection operator, proj, is computed in two steps. First, we find a joint approximating normal distribution via the Laplace approximation and then we project this onto each of the random variables $\mathbf{z}_u$ and $\mathbf{z}_v$. In the Laplace approximation step, we need to find the mode of $\log \xi(\mathbf{z}_u, \mathbf{z}_v)$ and calculate its Hessian at the mode to obtain the mean and variance of the approximating normal distribution:

$$\mu_{uv}^\xi = \underset{(\mathbf{z}_u, \mathbf{z}_v)}{\arg \max} \log \xi(\mathbf{z}_u, \mathbf{z}_v) \qquad (19)$$

$$\Sigma_{uv}^\xi = \left(\nabla^2_{(\mathbf{z}_u, \mathbf{z}_v) = \mu_{uv}^\xi} \log \xi(\mathbf{z}_u, \mathbf{z}_v) \right)^{-1}. \qquad (20)$$

The optimization problem in (19) is solved by optimizing first over $\mathbf{z}_u$ then over $\mathbf{z}_v$. The optimal value of $\mathbf{z}_u$ can be computed in closed form in terms of $\mathbf{z}_v$, since only normal densities are involved. Then the optimal value of $\mathbf{z}_v$ is found via gradient methods (e.g., BFGS). The function $\log \xi(\mathbf{z}_u, \mathbf{z}_v)$ is concave, so we can always find the global optimum. Note that this decomposition approach only depends on the tree structure of the model and hence will work for any observation distribution.

At the mode, we find the mean and variance of the normal distribution approximating $p(\mathbf{z}_u, \mathbf{z}_v | \mathbf{y})$ via (19) and (20). With this distribution, the edge counts can be inferred with the method of Section 3.2. In the projection step in (17) and (18), this distribution is projected to one of $\mathbf{z}_u$ or $\mathbf{z}_v$ by marginalizing out the other.

4.2. Complexity analysis

What is the computational complexity of inference with the GCGM? When inferring node counts, we must solve the optimization problem and compute a fixed number of matrix inverses. Each matrix inverse takes time $L^{2.5}$. In the Laplace approximation step, each gradient calculation takes $O(L^2)$ time. Suppose m iterations are needed. In the outer loop, suppose we must perform r passes of EP message passing and each iteration sweeps through the whole tree. Then the overall time is $O(r|E| \max(mL^2, L^{2.5}))$. The maximization problem in the Laplace approximation is smooth and concave, so it is relatively easy. In our experiments, EP usually converges within 10 iterations.

In the task of inferring edge counts, we only consider the complexity of calculating the mean, as this is all that is needed in our applications. This part is solved in closed form, with the most time-consuming operation being the matrix inversion. By exploiting the simple structure of the covariance matrix of $\mathbf{z}_{uv}$, we can obtain an inference method with time complexity of $O(L^3)$.

5. Experimental Evaluation

In this section, we evaluate the performance of our method and compare it to the MAP approximation of Sheldon, Sun, Kumar, and Dietterich (2013). The evaluation data are generated from the bird migration model introduced in Sheldon et al. (2013). This model simulates the migration of a population of M birds on an $L = \ell \times \ell$ map. The entire population is initially located in the bottom left corner of the map. Each bird then makes independent migration decisions for $T = 20$ time steps. The transition probability from cell i to cell j at each time step is determined by a logistic regression equation that employs four features. These features encode the distance from cell i to cell j, the degree to which cell j falls near the path from cell i to the destination cell in the upper right corner, the degree to which cell j lies in the direction toward which the wind is blowing, and a factor that encourages the bird to stay in cell i. Let $\mathbf{w}$ denote the parameter vector for this logistic regression formula. In this simulation, the individual model for each bird is a T-step Markov chain $X = (X_1, \ldots, X_{20})$ where the domain of each X_t consists of the L cells in the map. The CGM variables $\mathbf{n} = (\mathbf{n}_1, \mathbf{n}_{1,2}, \mathbf{n}_2, \ldots, \mathbf{n}_T)$ are vectors of length L containing counts of the number of birds in each cell at time t and the number of birds moving from cell i to cell j from time t to time $t + 1$. We will refer to these as the "node counts" (N) and the "edge counts" (E). At each time step t, the data generation model generates an observation vector $\mathbf{y}_t$ of length L which contains noisy counts of birds at all map cells at time t, $\mathbf{n}_t$. The observed counts are generated by a Poisson distribution with unit intensity.

We consider two inference tasks. In the first task, the parameters of the model are given, and the task is to infer the expected value of the posterior distribution over $\mathbf{n}_t$ for each time step t given the observations $\mathbf{y}_1, \ldots, \mathbf{y}_T$ (aka "smoothing"). We measure the accuracy of the node counts and edge counts separately.

An important experimental issue is that we cannot compute the true MAP estimates for the node and edge counts. Of course we have the values generated during the simulation, but because of the noise introduced into the observations, these are not necessarily the expected values of the posterior. Instead, we estimate the expected values by running the MCMC method (Sheldon & Dietterich, 2011) for a burn-in period of 1 million Gibbs iterations

and then collecting samples from 10 million Gibbs iterations and averaging the results. We evaluate the accuracy of the approximate methods as the relative error $||\mathbf{n}_{app} - \mathbf{n}_{mcmc}||_1/||\mathbf{n}_{mcmc}||_1$, where $\mathbf{n}_{app}$ is the approximate estimate and $\mathbf{n}_{mcmc}$ is the value obtained from the Gibbs sampler. In each experiment, we report the mean and standard deviation of the relative error computed from 10 runs. Each run generates a new set of values for the node counts, edge counts, and observation counts and requires a separate MCMC baseline run.

We compare our method to the approximate MAP method introduced by Sheldon et al. (2013). By treating counts as continuous and approximating the log factorial function, their MAP method finds the approximate mode of the posterior distribution by solving a convex optimization problem. Their work shows that the MAP method is much more efficient than the Gibbs sampler and produces inference results and parameter estimates very similar to those obtained from long MCMC runs.

The second inference task is to estimate the parameters $\mathbf{w}$ of the transition model from the observations. This is performed via Expectation Maximization, where our GCGM method is applied to compute the E step. We compute the relative error with respect to the true model parameters.

Table 1 compares the inference accuracy of the approximate MAP and GCGM methods. In this table, we fixed $L = 36$, set the logistic regression coefficient vector $\mathbf{w} = (1, 2, 2, 2)$, and varied the population size $N \in \{36, 360, 1080, 3600\}$. At the smallest population size, the MAP approximation is slightly better, although the result is not statistically significant. This makes sense, since the Gaussian approximation is weakest when the population size is small. At all larger population sizes, the GCGM gives much more accurate results. Note that the MAP approximation exhibits much higher variance as well.

Table 1. Relative error in estimates of node counts ("N") and edge counts ("E") for different population sizes N.

$N =$	36	360	1080	3600
MAP(N)	.173±.020	.066±.015	.064±.012	.069±.013
MAP(E)	.350±.030	.164±.030	.166±.027	.178±.025
GCGM(N)	.184±.018	.039±.007	.017±.003	.009±.002
GCGM(E)	.401±.026	.076±.008	.034±.003	.017±.002

Our second inference experiment is to vary the magnitude of the logistic regression coefficients. With large coefficients, the transition probabilities become more extreme (closer to 0 and 1), and the Gaussian approximation should not work as well. We fixed $N = 1080$ and $L = 36$ and evaluated three different parameter vectors: $\mathbf{w}_{0.5} = (0.5, 1, 1, 1)$, $\mathbf{w}_1 = (1, 2, 2, 2)$ and $\mathbf{w}_2 = (2, 4, 4, 4)$. Table 2 shows that for $\mathbf{w}_{0.5}$ and $\mathbf{w}_1$, the GCGM is much more accurate, but for $\mathbf{w}_2$, the MAP approximation gives a

slightly better result, although it is not statistically significant based on 10 trials.

Table 2. Relative error in estimates of node counts ("N") and edge counts ("E") for different settings of the logistic regression parameter vector $\mathbf{w}$

	$\mathbf{w}_{0.5}$	$\mathbf{w}_1$	$\mathbf{w}_2$
MAP(N)	.107±.014	.064±.012	.018±.004
MAP(E)	.293±.038	.166±.027	.031±.004
GCGM(N)	.013±.002	.017±.003	.024±.004
GCGM(E)	.032±.004	.034±.003	.037±.005

Our third inference experiment explores the effect of varying the size of the map. This increases the size of the domain for each of the random variables and also increases the number of values that must be estimated (as well as the amount of evidence that is observed). We vary $L \in \{16, 25, 36, 49\}$. We scale the population size accordingly, by setting $N = 30L$. We use the coefficient vector $\mathbf{w}_1$. The results in Table 3 show that for the smallest map, both methods give similar results. But as the number of cells grows, the relative error of the MAP approximation grows rapidly as does the variance of the result. In comparison, the relative error of the GCGM method barely changes.

Table 3. Relative inference error with different map size

$L =$	16	25	36	49
MAP(N)	.011±.005	.025±.007	.064±.012	.113±.015
MAP(E)	.013±.004	.056±.012	.166±.027	.297±.035
GCGM(N)	.017±.003	.017±.003	.017±.003	.020±.003
GCGM(E)	.024±.002	.027±.003	.034±.003	.048±.005

We now turn to measuring the relative accuracy of the methods during learning. In this experiment, we set $L = 16$ and vary the population size for $N \in \{16, 160, 480, 1600\}$. After each EM iteration, we compute the relative error as $||\mathbf{w}_{learn} - \mathbf{w}_{true}||_1/||\mathbf{w}_{true}||_1$, where $\mathbf{w}_{learn}$ is the parameter vector estimated by the learning methods and $\mathbf{w}_{true}$ is the parameter vector that was used to generate the data. Figure 1 shows the training curves for the three parameter vectors $\mathbf{w}_{0.5}, \mathbf{w}_1$, and $\mathbf{w}_2$. The results are consistent with our previous experiments. For small population sizes ($N = 16$ and $N = 160$), the GCGM does not do as well as the MAP approximation. In some cases, it overfits the data. For $N = 16$, the MAP approximation also exhibits overfitting. For $\mathbf{w}_2$, which creates extreme transition probabilities, we also observe that the MAP approximation learns faster, although the GCGM eventually matches its performance with enough EM iterations.

Our final experiment measures the CPU time required to perform inference. In this experiment, we varied $L \in \{16, 36, 64, 100, 144\}$ and set $N = 100L$. We used parameter vector $\mathbf{w}_1$. We measured the CPU time consumed to infer the node counts and the edge counts. The MAP
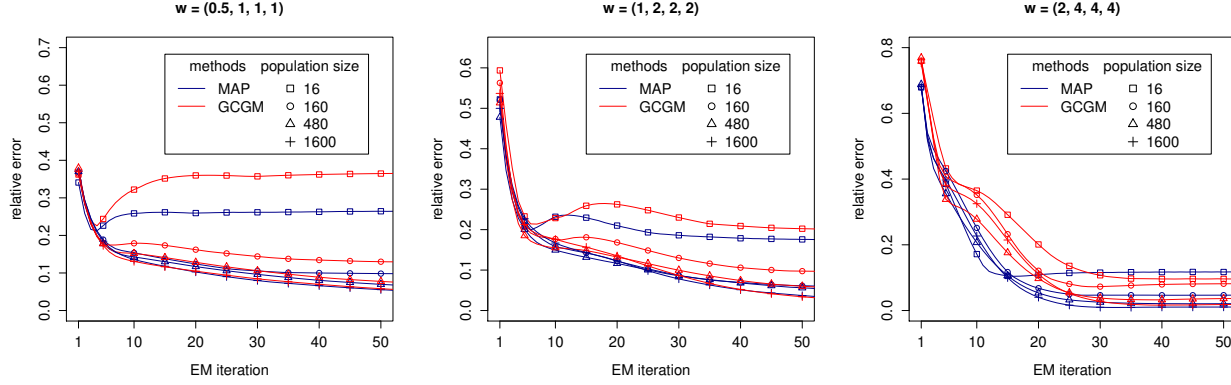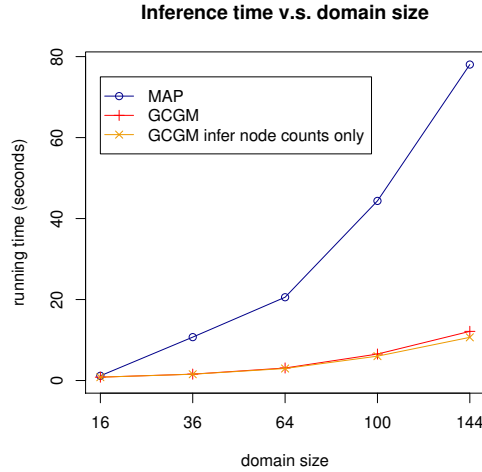
Figure 1. EM convergence curve different feature coefficient and population sizes

Figure 2. A comparison of inference run time with different numbers of cells L

method infers the node and edge counts jointly, whereas the GCGM first infers the node counts and then computes the edge counts from them. We report the time required for computing just the node counts and also the total time required to compute the node and edge counts. Figure 2 shows that the running time of the MAP approximation is much larger than the running time of the GCGM approximation. For all values of L except 16, the average running time of GCGM is more than 6 times faster than for the MAP approximation. The plot also reveals that the computation time of GCGM is dominated by estimating the node counts. A detailed analysis of the implementation indicates that the Laplace optimization step is the most time-consuming.

In summary, the GCGM method achieves relative error that matches or is smaller than that achieved by the MAP ap-

proximation. This is true both when measured in terms of estimating the values of the latent node and edge counts and when estimating the parameters of the underlying graphical model. The GCGM method does this while running more than a factor of 6 faster. The GCGM approximation is particularly good when the population size is large and when the transition probabilities are not near 0 or 1. Conversely, when the population size is small or the probabilities are extreme, the MAP approximation gives better answers although the differences were not statistically significant based on only 10 trials. A surprising finding is that the MAP approximation has much larger variance in its answers than the GCGM method.

6. Concluding Remarks

This paper has introduced the Gaussian approximation (GCGM) to the Collective Graphical Model (CGM). We have shown that for the case where the observations only depend on the separators, the GCGM is the limiting distribution of the CGM as the population size $N \rightarrow \infty$. We showed that the GCGM covariance matrix maintains the conditional independence structure of the CGM, and we presented a method for efficiently inverting this covariance matrix. By applying expectation propagation, we developed an efficient algorithm for message passing in the GCGM with non-Gaussian observations. Experiments on a bird migration simulation showed that the GCGM method is at least as accurate as the MAP approximation of Sheldon et al. (2013), that it exhibits much lower variance, and that it is 6 times faster to compute.

Acknowledgement

References

Dawid, A. P. and Lauritzen, S. L. Hyper Markov laws in the statistical analysis of decomposable graphical models. *The Annals of Statistics*, 21(3):1272–1317, 1993.

Feller, W. *An Introduction to Probability Theory and Its Applications, Vol. 2*. Wiley, 1968.

Lauritzen, S.L. *Graphical models*. Oxford University Press, USA, 1996.

Loh, Po-Ling and Wainwright, Martin J. Structure estimation for discrete graphical models: Generalized covariance matrices and their inverses. *The Annals of Statistics*, 41(6):3022–3049, 2013.

Sheldon, Daniel and Dietterich, Thomas G. Collective Graphical Models. In *NIPS 2011*, 2011.

Sheldon, Daniel, Sun, Tao, Kumar, Akshat, and Dietterich, Thomas G. Approximate Inference in Collective Graphical Models. In *Proceedings of ICML 2013*, pp. 9, 2013.

Sundberg, R. Some results about decomposable (or Markov-type) models for multidimensional contingency tables: distribution of marginals and partitioning of tests. *Scandinavian Journal of Statistics*, 2(2):71–79, 1975.

Wainwright, M.J. and Jordan, M.I. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

Ypma, Alexander and Heskes, Tom. Novel approximations for inference in nonlinear dynamical systems using expectation propagation. *Neurocomputing*, 69(1-3):85–99, 2005.

A. Proof of Proposition 1

The usual way of writing the CGM distribution is to replace $f(\mathbf{n}; \boldsymbol{\theta})$ in Eq. (3) by

$$f'(\mathbf{n}; \boldsymbol{\theta}) = \frac{\prod_{C \in \mathcal{C}, i_C \in \mathcal{X}^{|C|}} \mu_C(i_C)^{\mathbf{n}_C(i_C)}}{\prod_{S \in \mathcal{S}, i_S \in \mathcal{X}^{|S|}} \left(\mu_S(i_S)^{\mathbf{n}_S(i_S)}\right)^{\nu(S)}} \quad (21)$$

We will show that $f(\mathbf{n}; \boldsymbol{\theta}) = f'(\mathbf{n}; \boldsymbol{\theta})$ for any $\mathbf{n}$ such that $h(\mathbf{n}) > 0$ by showing that both descibe the probability of an ordered sample with sufficient statistics $\mathbf{n}$. Indeed, suppose there exists some ordered sample $\mathbf{X} = (\mathbf{x}^1, \dots, \mathbf{x}^N)$ with sufficient statistics $\mathbf{n}$. Then it is clear from inspection of Eq. (3) and Eq. (21) that $f(\mathbf{n}; \boldsymbol{\theta}) = \prod_{m=1}^{N} p(\mathbf{x}^m; \boldsymbol{\theta}) = f'(\mathbf{n}; \boldsymbol{\theta})$ by the junction tree reparameterization of $p(\mathbf{x}; \boldsymbol{\theta})$ (Wainwright & Jordan, 2008). It only remains to show that such an $\mathbf{X}$ exists whenever $h(\mathbf{n}) > 0$. This is exactly what was shown by Sheldon & Dietterich (2011): for junction trees, the hard constraints of Eq. (4), which enforce local consistency on the integer count variables, are equivalent to the global consistency property that there exists some ordered sample $\mathbf{X}$ with sufficient statistics equal to $\mathbf{n}$. (Since these are integer count variables, the proof is quite different from the similar theorem that local consistency implies global consistency for marginal distributions.) We briefly note two interesting corollaries to this argument. First, by the same reasoning, *any* reparameterization of $p(\mathbf{x}; \boldsymbol{\theta})$ that factors in the same way can be used to replace $f(\mathbf{n}; \boldsymbol{\theta})$ in the CGM distribution. Second, we can see that the base measure $h(\mathbf{n})$ is exactly the *number of different ordered samples* with sufficient statistics equal to $\mathbf{n}$.

B. Proof of Theorem 1: Additional Details

Suppose $\{\mathbf{n}^N\}$ is a sequence of random vectors that converge in distribution to $\mathbf{n}$, and that $\mathbf{n}_A^N$, $\mathbf{n}_B^N$, and $\mathbf{n}_S^N$ are subvectors that satisfy

$$\mathbf{n}_A^N \perp\!\!\!\perp \mathbf{n}_B^N \mid \mathbf{n}_S^N \quad (22)$$

for all N. Let α, β, and γ be measurable sets in the appropriate spaces and define

$$z = \Pr(\mathbf{n}_A \in \alpha, \mathbf{n}_B \in \beta \mid \mathbf{n}_S \in \gamma) - \quad (23)$$
$$\Pr(\mathbf{n}_A \in \alpha \mid \mathbf{n}_S \in \gamma) \Pr(\mathbf{n}_B \in \beta \mid \mathbf{n}_S \in \gamma)$$

Also let z^N be the same expression but with all instances of $\mathbf{n}$ replaced by $\mathbf{n}^N$ and note that $z^N = 0$ for all N by the assumed conditional independence property of Eq. (22). Because the sequence $\{\mathbf{n}^N\}$ converges in distribution to $\mathbf{n}$, we have convergence of each term in z^N to the corresponding term in z, which means that

$$z = \lim_{N \to \infty} z^N = \lim_{N \to \infty} 0 = 0,$$

so the conditional independence property of Eq. (22) also holds in the limit.

C. Proof of Theorem 3: Linear Function from $\tilde{\mathbf{I}}$ to $\mathbf{I}$

We need to show $\mathbf{I}_A$ can be recovered from $\tilde{\mathbf{I}}_{A+}$ with a linear function.

Suppose the last indicator variable in $\mathbf{I}_A$ is i_A^0, which corresponds to the setting that all nodes in A take value L. Let $\mathbf{I}_A'$ be a set of indicators which contains all entries in $\mathbf{I}_A$ but the last one i_A^0. Then $\mathbf{I}_A$ can be recovered from $\mathbf{I}_A'$ by the constraint $\sum_{i_A} \mathbf{I}_A(i_A) = 1$.

Now we only need to show that $\mathbf{I}_A'$ can be recovered from $\mathbf{I}_{A+}$ linearly. We claim that there exists an invertible matrix $\mathbb{H}$ such that $\mathbb{H} \, \mathbf{I}_A' = \tilde{\mathbf{I}}_{A+}$.

Showing the existence of $\mathbb{H}$. Let $\tilde{\mathbf{I}}_{A+}(i_D)$ be the i_D entry of $\tilde{\mathbf{I}}_{A+}$, which is for configuration i_D of clique $D, D \subseteq A$.

$$\tilde{\mathbf{I}}_{A+}(i_D) = \sum_{i_{A \setminus D}} \mathbf{I}_A'(i_D, i_{A \setminus D}) \quad (24)$$

Since no nodes in D take value L by definition of $\tilde{\mathbf{I}}_D$, $(i_D, i_{A \setminus D})$ *cannot* be the missing entry i_A^0 of $\mathbf{I}_A'$, and the equation is always valid.

Showing that $\mathbb{H}$ is square. For each D, there are $(L-1)^{|D|}$ entries, and A has $\binom{|A|}{|D|}$ sub-cliques with size $|D|$. So $\tilde{\mathbf{I}}_{A+}$ have overall $L^{|A|} - 1$ entries, which is the same as $\mathbf{I}_A'$. So $\mathbb{H}$ is a square matrix.

We view $\mathbf{I}_A'$ and $\tilde{\mathbf{I}}_{A+}$ as matrices and each row is a indicator function of graph configurations. Since no trivial linear combination of $\tilde{\mathbf{I}}_{A+}$ is a constant by the conclusion in Loh and Wainwright (2013), $\tilde{\mathbf{I}}_{A+}$ has linearly independent columns. Therefore, $\mathbb{H}$ must have full rank and $\mathbf{I}_A'$ must have linearly independent columns.