Published in final edited form as:

Wiley Interdiscip Rev Comput Stat. 2013 ; 5(6): 478–485. doi:10.1002/wics.1273.

Gibbs ensembles for incompatible dependency networks

Shyh-Huei Chen^{1,*}, Edward H. Ip¹, and Yuchung J. Wang²

¹Department of Biostatistical Sciences, Wake Forest University School of Medicine, Winston-Salem, NC, USA

²Department of Mathematical Sciences, Rutgers University, Camden, NJ, USA

Abstract

In most statistical applications, the Gibbs sampler is the method of choice for inference regarding conditionally specified distributions that are compatible. Compatibility ensures that a unique Gibbs distribution exists. For machine learning of complex models such as dependency networks, the conditional models are sometimes incompatible. In this paper, we review an ensemble approach using the Gibbs sampler as the base procedure. A Gibbs ensemble consists of many joint distributions resulting from different scan orders of the same conditional model, and the solution is a weighted sum of the ensemble. The algorithm is scalable and can handle large data sets of high dimensionality. The proposed approach provides joint distributions that conform with the conditional specifications better than the solutions obtained by linear programming and by a fixed-scan Gibbs sampler alone. Owing to incompatibility, the invariant distribution of a Gibbs sampler is scan-order dependent. A Gibbs ensemble is the collection of joint distributions estimated from the Gibbs samples of different scan orders.

Keywords

Gibbs sampler; conditionally specified distribution; dependency network; linear programming; ensemble method

CONDITIONALLY SPECIFIED DISTRIBUTIONS

A reduced model for multivariate data may be conveniently formulated in two ways: as an undirected graphical model or as an acyclic, directed graphical model. An undirected graphical model describes all of the variables jointly in a product of potential functions,¹ while an acyclic-directed graphical model factors the joint distribution into a serial product of conditional densities. Arnold et al.² reviewed a more intricate approach: statistically modeling the joint with a collection of conditional distributions, which they called conditionally specified distributions (CSDs). Figure 1 shows several representations of reduced models of three variables— x_1 (age), x_2 (gender), and x_3 (income). In Figure 1(a), the joint density is written as $p(x_1, x_2, x_3) = p(x_3|x_1, x_2)p(x_1, x_2)$, and in Figure 1(b), the

*Correspondence to: schen@wakehealth.edu.

Additional Supporting Information may be found in the online version of this article.

Conflict of interest: The authors have declared no conflicts of interest for this article.

joint density is determined collectively by $p(x_1|x_2, x_3)$, $p(x_2|x_3)$, and $p(x_3|x_1, x_2)$. Because the arrows—which indicate the flow of information—form a circle, a CSD is represented by the cyclic, directed graph shown in Figure 1(b). Figure 1(c), on the other hand, represents an undirected graph.

Table 1 lists the different terminologies. The community of machine learning regards a graph as a network, while statisticians tend to emphasize the mode of representation. Both Markov networks and Bayesian networks take no issue of compatibility thus learning their distributions directly from the data. For a domain of $\mathbf{N} = \{x_1, \dots, x_J\}$, let $\text{pa}(x_j)$ represent the parent nodes of x_j and $\text{pa}(x_j) \subset \{x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_J\}$. The local probability distribution of x_j is

$$\begin{aligned} p_j(x_j|\text{pa}(x_j)) &= p_j(x_j|x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_J) \\ &= p_j(x_j|\mathbf{N} \setminus x_j). \end{aligned}$$

A dependency network is a pair $(\mathcal{G}, \mathcal{P})$ where $\mathcal{G}$ is a cyclic graph of nodes $x_j, j = 1, \dots, J$, and $\mathcal{P}$ is the collection of local distributions $p(x_j|\mathbf{N} \setminus x_j), j = 1, \dots, J$. In many applications, $p(x_j|\mathbf{N} \setminus x_j)$ is either a regression or a classifier. Data are used to learn the local distributions. Then, for the purpose of making a probability inference among nodes, a joint distribution needs to be formulated from $\mathcal{P}$. Such a local specification of a global distribution was first proposed by Besag³ for modeling spatial observations in which the parent nodes are the spatially adjacent nodes; thus, the overall distribution is a reduced model. Hereinafter, the terms CSD and dependency network will be used interchangeably.

Issues of Conditionally Specified Distribution

Like a system of equations, CSD must deal with the questions of the existence, uniqueness, and computation/approximation of a solution. The three issues of CSD include:

1. Compatibility: A local specification $\mathcal{P}$ is called strongly compatible if there exists (almost everywhere) a unique $p(x_1, \dots, x_J)$ such that

$$p_j(x_j|\mathbf{N} \setminus x_j) = p(x_1, \dots, x_J) / \int p(x_1, \dots, x_J) dx_j.$$
2. Sufficiency: Does a compatible $\mathcal{P}$ have sufficient information to define a joint distribution?
3. Computation: What are computationally feasible ways to approximate a joint distribution given $\mathcal{P}$ that is both compatible and sufficient?

Tierney⁴ stated that ‘strong compatibility is an obvious requirement for the Gibbs chain to be positively recurrent’, which is a necessary condition for convergence to a unique stationary distribution. When $\mathcal{P}$ are compatible, Heckerman et al.⁵ called the dependency network consistent. Arnold and Press⁶ were among the first to show ways to assure compatibility, and they also proposed the use of the Gibbs sampler for computation. When the CSDs are not necessarily compatible, the joint distribution can still be well approximated with a sufficiently large number of observations. Regardless, it can be difficult to know the

sample size needed for a close approximation. Sufficiency will not be discussed in this review. Instead, we refer the readers to Ref 7.

Compatible Dependency Networks

The problem of characterizing a joint distribution by a collection of compatible CSDs has been extensively studied in the past two decades (see Refs ^{2, 3, 6–9}. For more recent development, see Refs ^{10–13}).

For the simplest case, compatibility is equivalent to the existence of densities $f(x_1)$ and $g(x_2)$ such that $p_1(x_1|x_2)g(x_2) = p_2(x_2|x_1)f(x_1)$. For discrete x_1 and x_2 , $f(x_1)$ and $g(x_2)$ can be obtained by solving a system of over-specified linear equations,⁶ but this approach has limitations. For example, the solution is not highly scalable. Ip and Wang¹² used interactions of the local distributions to verify compatibility. Their approach is general as long as the variables are variation independent. In a similar spirit, Wang and Ip¹⁰ proved that differentiable $\mathcal{P}$ are compatible provided that

$$\frac{\partial^2}{\partial x_i \partial x_j} \log p_j(x_j | \mathcal{N} \setminus x_j) = \frac{\partial^2}{\partial x_i \partial x_j} \log p_i(x_i | \mathcal{N} \setminus x_i). \quad (1)$$

This result showed that compatibility is determined by the differential properties of the local distributions, and no integration needs to be invoked. Moreover, Wang and Ip¹⁰ and Ip and Wang¹² proved that $\mathcal{P}$ satisfies sufficiency provided that

$$p_j(x_j | \text{pa}(x_j)) = p_j(x_j | \mathcal{N} \setminus x_j).$$

Otherwise, $\mathcal{P}$ can have multiple joint distributions. Both papers also showed a straightforward way to construct a closed-form joint distribution using selected interactions of the local distributions. However, for approximating the joint, the Gibbs sampler is often the method of choice because the codes of implementation are readily available.

Incompatible Dependency Networks

Heckerman et al.⁵ pointed out that ‘it is computationally expensive to learn a Markov network from data and then convert that network to a consistent dependency network’. Instead, they proposed using multiple approaches to estimate the local distributions from data. Once all $p_j(x_j | \text{pa}(x_j))$, $j=1, \dots, J$, have been estimated, the next step is to construct a joint from the dependency encoded in these estimates. Because the CSDs $p_j(x_j | \text{pa}(x_j))$ are obtained independently and using different algorithms, they are not necessarily compatible. Hence, in general, there does not exist a joint distribution from which every local distribution can be derived. Technically speaking, the procedure of applying a Gibbs sampling to an incompatible $\mathcal{P}$ is not a true Gibbs sampler because the existence of a unique solution is not guaranteed. Heckerman et al.⁵ termed it the pseudo-Gibbs sampler. Although empirically in many cases the Gibbs samplers based on an incompatible $\mathcal{P}$ converge, there is little research on the adequacy or the performance of such solutions. Under a hypothetical assumption that incompatible $\mathcal{P}$ will be ‘almost consistent when learned from data with a

large sample size', Heckerman et al.⁵ used an ordered-scan pseudo-Gibbs sampler (to be explained later) to produce a joint distribution. The solution is likely to be inconsistent with many of the local distributions, but as argued by the authors, it 'should be almost consistent'.

In statistics, Arnold and Gokhale¹⁴ proposed using nonlinear programming (with a minimal Kullback-Kiebler measure) and Arnold et al.¹⁵ proposed linear programming for finding the most nearly compatible joint distribution for a given collection of CSDs. For example, given the two-dimensional (discrete) conditional distributions with $p_1(x_1 = i | x_2 = j) = a_{ij}$ and $p_2(x_2 = j | x_1 = i) = b_{ij}$, the algorithm found the smallest $\varepsilon \geq 0$ and $m_{ij} \geq 0$ such that the following inequalities hold:

$$|a_{ij} \sum_i m_{ij} - m_{ij}| \leq \varepsilon \text{ and } |b_{ij} \sum_j m_{ij} - m_{ij}| \leq \varepsilon, \forall i, j.$$

If $\varepsilon = 0$, then p_1 and p_2 are compatible, with m_{ij} being the joint distribution. On the other hand, if $\varepsilon > 0$, then p_1 and p_2 are ε -compatible. Linear or nonlinear programming requires expertise and training, and new codes will be needed when the objective function or the number of nodes is changed.

GIBBS ENSEMBLE

Chen et al.¹⁶ proposed an ensemble approach using the pseudo-Gibbs sampler as the base procedure. In principle, an ensemble is a collection of reasonable but not necessarily optimal solutions. After the ensemble is formed, a linear combination could improve upon every solution in the ensemble. Boosting¹⁷ and bagging¹⁸ are two such algorithms, and they are relatively simple devices to improve the predictive power of base procedures. Using power divergence as the objective function, the Gibbs ensemble is an unsupervised learning algorithm that improves the pseudo-Gibbs sampler. The members of the Gibbs ensemble are learned from $\mathcal{P}$ by way of different scan orders.

Consider a collection of CSDs of J random variables $\{x_1, \dots, x_J\}$ whose model is specified by $\mathcal{P} = \{p_j(x_j | \mathbb{N} \setminus x_j)\}$. Let S_J be the symmetric group of all possible permutations of $(1, \dots, J)$ and $\|S_J\| = J!$. For every $(k_1, \dots, k_J) \in S_J$, arrange the given local distributions of $\mathcal{P}$ in the following sequence:

$$p_{k_1} \rightarrow p_{k_2} \rightarrow \dots \rightarrow p_{k_J} \rightarrow p_{k_1} \rightarrow \dots \dots,$$

and generate samples using a pseudo-Gibbs sampler in this order. Every $(k_1, \dots, k_J) \in S_J$ represents a *scan pattern* or a scan order. A Gibbs sampler in the above setting with a specified scan pattern is called a fixed-scan Gibbs sampler, as opposed to scanning $\mathcal{P}$ in a random order with the prescribed probabilities, or a random-scan Gibbs sampler. The Gibbs ensemble procedure can be described as follows.

Begin with a randomly selected $X^{(0)} = (x_1^0, \dots, x_J^0)$. Conditioned on $(x_i^0, i \in \mathbb{N} \setminus x_{k_1}^0)$ draw an $x_{k_1}^{(1)}$ from p_{k_1} and update $X^{(0)}$ to $X^{(1)} = (X^{(0)} \setminus x_{k_1}^{(0)}) \cup x_{k_1}^{(1)}$. That is, $X^{(1)}$ differs from $X^{(0)}$

only in the k_1 component. Next, conditioned on $X^{(1)} \setminus x_{k_2}^{(1)}$, draw an $x_{k_2}^{(2)}$ from p_{k_2} and form $X^{(2)}$ by replacing the k_2 component on $X^{(1)}$ with the updated value. Successive sampling steps from p_{k_1} to p_{k_J} are called a cycle, and, after one cycle, every x_i^0 of $X^{(0)}$ will be updated once. Let $X^{(J)}$ be the result at the end of the first cycle. After w burn-in cycles, $X^{(Jw+1)}$ is harvested at the end of the next cycle as the first qualified sample, and the cycle is repeated l times. Let the empirical distribution of $\{X^{(hJw+1)}, 1 \leq h \leq l\}$ be $\hat{\pi}^{(k_1, \dots, k_J)}$, where the superscript indicates the scan pattern. Every $\hat{\pi}^{(k_1, \dots, k_J)}$ is called a *Gibbs distribution*.^{19,20}

The collection of $J!$ such Gibbs distributions, $\{\hat{\pi}^{(k_1, \dots, k_J)}, (k_1, \dots, k_J) \in S_J\}$, is named the *Gibbs ensemble*, after its base procedure. To compute the error of each Gibbs distribution, convert each $\hat{\pi}^{(k_1, \dots, k_J)}$ to the collection of conditional distributions

$\hat{\pi}_j^{(k_1, \dots, k_J)}(x_j | \mathbb{N} \setminus x_j), 1 \leq j \leq J$, and measure the error of $\hat{\pi}_j^{(k_1, \dots, k_J)}(x_j | \mathbb{N} \setminus x_j)$ from the corresponding p_j via a divergence measure. The sum of J divergences is the error of $\hat{\pi}^{(k_1, \dots, k_J)}$. In theory, the error is 0 if and only if $\mathcal{P}$ is compatible; however, in simulation, errors of a collection of compatible CSDs will not be exactly 0 due to Monte Carlo variations.

Several commonly used divergences $D(\hat{\mathbf{p}}, \mathbf{p})$ between two distributions $\hat{\mathbf{p}}$ and $\mathbf{p}$ are [e.g.,

Ref 21 p. 348–349]: $L^2 := \sum (\hat{p}_i - p_i)^2$ (Euclidean); $F^2 := 4 \sum (\sqrt{\hat{p}_i} - \sqrt{p_i})^2$ (Freeman-Turkey); $X^2 := \sum (\hat{p}_i - p_i)^2 / \hat{p}_i$ (Pearson's chi-square); $N^2 := \sum (\hat{p}_i - p_i)^2 / p_i$ (Neyman's chi-square); $I^2 := \sum \hat{p}_i \log(\hat{p}_i / p_i)$ (Information); and $G^2 := \sum p_i \log(p_i / \hat{p}_i)$ (Kullback–Leibler).

Let $e_D^{(k_1, \dots, k_J)}$ denote the $D(\hat{\mathbf{p}}, \mathbf{p})$ between $\hat{\pi}^{(k_1, \dots, k_J)}$ and $\mathcal{P}$; that is,

$$e_D^{(k_1, \dots, k_J)} = \sum_{i=1}^J D\left(\hat{\pi}_i^{(k_1, \dots, k_J)}; p_i\right). \text{ Also, let } E_D = \sum (1/e_D^{(k_1, \dots, k_J)}), \text{ and let}$$

$\varepsilon_D^{(k_1, \dots, k_J)} = \left(1/e_D^{(k_1, \dots, k_J)}\right) / E_D$, which is inversely proportional to the error, be the weight assigned to $\hat{\pi}^{(k_1, \dots, k_J)}$. The ensemble solution, termed the *Gibbs ensemble (or GE) solution*, is

$$\tilde{\pi}_D = \sum_{(k_1, \dots, k_J) \in S_J} \varepsilon_D^{(k_1, \dots, k_J)} \hat{\pi}^{(k_1, \dots, k_J)}. \quad (2)$$

For example, $\tilde{\pi}_L$ is the GE solution weighted by the L^2 -divergence. Figure 2 illustrates the steps for obtaining the GE solution.

For $J=2$, let p_1 and p_2 be the local distributions of $(x_1|x_2)$ and $(x_2|x_1)$, respectively. The Gibbs ensemble contains two distributions: $\hat{\pi}^{(1,2)}$ and $\hat{\pi}^{(2,1)}$. The distribution $\hat{\pi}^{(1,2)}$ is the empirical distribution of $\left\{\left(x_1^{(2hw+1)}, x_2^{(2hw+2)}\right), 1 \leq h \leq l\right\}$, sampled from the following sequence:

$$\begin{aligned} \left(\cdot | x_2^{(0)} \right) &\xrightarrow{p_1} \left(x_1^{(1)} | x_2^{(0)} \right) \xrightarrow{p_2} \left(x_2^{(1)} | x_2^{(1)} \right) \\ &\xrightarrow{p_1} \left(x_1^{(2)} | x_2^{(1)} \right) \xrightarrow{p_2} \dots \end{aligned}$$

On the other hand, $\hat{\pi}^{(2,1)}$ is based on samples from the $p_2 \rightarrow p_1 \rightarrow \dots p_2 \rightarrow p_1 \rightarrow \dots$ sequence. The GE is $\tilde{\pi}_D = \varepsilon_D^{(1,2)} \hat{\pi}^{(1,2)} + \varepsilon_D^{(2,1)} \hat{\pi}^{(2,1)}$.

For $J=3$, there are six Gibbs distributions, and the GE solution is the weighted average of these distributions:

$$\begin{aligned} \tilde{\pi}_D = & \varepsilon_D^{(1,2,3)} \hat{\pi}^{(1,2,3)} + \varepsilon_D^{(1,3,2)} \hat{\pi}^{(1,3,2)} + \varepsilon_D^{(2,1,3)} \hat{\pi}^{(2,1,3)} \\ & + \varepsilon_D^{(2,3,1)} \hat{\pi}^{(2,3,1)} + \varepsilon_D^{(3,1,2)} \hat{\pi}^{(3,1,2)} + \varepsilon_D^{(3,2,1)} \hat{\pi}^{(3,2,1)}. \end{aligned}$$

EXAMPLE FOR DISCRETE RANDOM VARIABLES

Consider the following CSDs (Ref 15, Example 10):

$$p_1 = \begin{pmatrix} \frac{1}{4} & \frac{1}{3} \\ \frac{3}{4} & \frac{2}{3} \end{pmatrix}, \text{ and } p_2 = \begin{pmatrix} \frac{1}{10} & \frac{2}{10} \\ \frac{9}{10} & \frac{8}{10} \end{pmatrix}.$$

Here, p_1 and p_2 correspond to $(x_1|x_2)$ and $(x_2|x_1)$, respectively. Arnold et al.¹⁵ obtained the following ε -optimal ($\varepsilon = 23/734$) joint distribution via linear programming:

$$\hat{\pi}_{LP} = \frac{1}{734} \begin{pmatrix} 55 & 179 \\ 73 & 427 \end{pmatrix}.$$

Table 2 compares the divergences obtained through different algorithms. Here, $\hat{\pi}_{RS}$ and $\hat{\pi}_{EQ}$ are for the random-scan Gibbs distribution (with equal-selection probability) and the equally weighted GE solution, respectively, whereas $\hat{\pi}_L$, $\hat{\pi}_P$, $\hat{\pi}_X$, and $\hat{\pi}_F$ are GE solutions using optimal weights based, respectively, on L^2 , P^2 , X^2 , and F^2 . The $\hat{\pi}_{LP}$ is slightly better than $\hat{\pi}_{RS}$ and $\hat{\pi}_{EQ}$ but worse than other weighted GE solutions, and the reduction of discrepancy of the GE solutions over the $\hat{\pi}_{LP}$ range from 5 to 22%, which is a clear indication that weights can play a critical role in improving the solution.

EXAMPLE FOR CONTINUOUS RANDOM VARIABLES

Consider the following incompatible CSDs with Gaussian conditionals:

$$\begin{aligned} p_1(x_1|x_2) &= N(x_2/2 + x_2^2/5, 1) \quad \text{and} \\ p_2(x_2|x_1) &= N(x_1/2, 1). \end{aligned}$$

We ran a GE solution using various weighting schemes and divergence measures. The first 1000 samples were treated as burn-in cycles and deleted, and the results were based on

100,000 samples. Figure 3 shows contour plots of $\hat{\pi}^{(1,2)}$, $\hat{\pi}^{(2,1)}$, and one GE solution $\tilde{\pi}_L$. The Gibbs sampler was implemented in the R language,²² and the R library KernSmooth²³ was adopted for kernel-smoothing estimation.

Table 3 compares the L^2 and F^2 divergences obtained through different methods: the fixed-scan Gibbs sampler, the equal-weight Gibbs ensemble, and the GE solutions $\tilde{\pi}_L$ and $\tilde{\pi}_F$. Again, the weighted average of Gibbs distributions generally performed better than the fixed-scan Gibbs distributions $\hat{\pi}^{(1,2)}$ and $\hat{\pi}^{(1,1)}$. The results suggest that the Gibbs ensemble is superior to a Gibbs sampler of single, fixed-scan order.

EXAMPLE OF BRAIN TUMOR DATA

The data set contained information from $N = 100$ patients with either a primary brain tumor or brain metastases. Specific details of the data set were reported elsewhere.²⁴ Here we only used several self-reported quality of life (qol) measures extracted from the functional assessment of cancer therapy (FACT) and the brain-tumor-specific subscale (FACT-B), both of which have been validated in other studies,^{25,26}. To illustrate the different approaches of the dependency networks, we used data collected from 10 qol items and examined the pattern of qol concerns and symptoms. The item responses were dichotomized as 1 = presence of symptom and 0 = absence of the concern or symptom.

Logistic regressions were applied to the data set such that each individual variable was treated as a dependent variable and other variables were treated as predictors. Backward selection was used to select the relevant predictors. The resulting dependency network is shown in Figure 4 and the keys of the items are listed in the Supporting Information. Because some structure zeros were present in the CSDs, only three divergences, L^2 , F^2 , and X^2 were reported.

It is impractical—and often not necessary—to include all permutations of scan patterns in the Gibbs ensemble. We modified the full-fledged GE solution and adopted the following ‘most promising subset’ procedure to arrive at a GE solution: (1) randomly select S ($S < J!$) scan patterns (or include prescribed scan patterns as suggested in Ref 5) to generate S pseudo-Gibbs solutions (PGSs) and (2) use s ($s \leq S$) PGSs to form an ensemble, which is called the s -nested Gibbs ensemble, and then select the one that is optimal in terms of its divergence. Specifically, in step (2) first rank the order of the PGSs in terms of their divergences and then form ensembles by adding PGSs one at a time, starting with the PGS of the lowest divergence, and following the rank order. The s -nested ensembles were then evaluated using a divergence criterion.

To illustrate this practical implementation of the GE, we used 10 randomly selected scan patterns to run the pseudo-Gibbs sampler, each with 1000 burn-in cycles and 1,000,000 sample points. The divergences with respect to conditional models for the PGSs and s -nested GE solutions are respectively presented in the upper and lower panel of Table 4. For example, the s_1 ensemble contains only PGS3—the PGS with the smallest divergence, s_2 contains PGS3 and PGS9, and s_{10} contains all 10 PGSs. Note that using all 10 PGSs does not necessarily perform better than the best single PGS. However, for the best s -nested GE

solutions, $s_4 = \{\text{PGS3, PGS9, PGS4, PGS6}\}$ (for L^2), $s_3 = \{\text{PGS3, PGS9, PGS4}\}$ (F^2), and $s_4 = \{\text{PGS3, PGS9, PGS4, PGS6}\}$ (X^2), the reductions in divergence are approximately 35, 40, and 37% in their respective divergence measures when compared to the best single PGS.

CONCLUSION

For high-dimensional data, a reduced model may be formulated in at least two ways: as an undirected graphical model—also known as a Markov network—in which all variables are considered jointly, or more intricately, as a system of univariate conditional models called a dependency network, which can be depicted by a cyclic, directed graph.⁵ In the second approach, as observed by Dobra et al.,²⁷ the conditionally specified models almost surely do not cohere to a proper joint distribution. That is, any dependency network is most likely to be incompatible. No general solution exists for the joint distribution in such a case. While not all data applications would need to derive a joint distribution—e.g., when prediction is the focus, there are scenarios under which the joint distribution is desired. One example is when a low-dimensional marginal distribution is a target. Under this kind of circumstances, it is of interest to compare the joint distributions obtained through different algorithms. In this review, we compared the Gibbs distributions from ordered, fixed scans, from the joint distributions obtained from linear programming, and from the GE solutions. We make the following observations:

1. The Gibbs ensemble improves upon individual Gibbs distributions of the ensemble. This is generally true for all ensemble algorithms, regardless of the criterion of divergence. Therefore, the Gibbs ensemble should be preferred over a single, ordered, fixed-scan Gibbs sampler for incompatible dependency networks. Nevertheless, based on our experience the F^2 measure appeared to be more stable and did not have a problem with zero cells.
2. The Gibbs sampler is both robust and scalable for handling high-dimensional dependency networks. Both the Gibbs sampler and the GE solution are easy to program. If these are important considerations for learning in a dependency network, then the Gibbs ensemble should be preferred over linear programming.
3. The Gibbs ensemble offers a computationally feasible approach—scan patterns can be run in parallel and the ensemble can be formed by the most promising subset—for obtaining optimal or nearly optimal solutions for incompatible dependency networks.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

This work was supported by NIH grants 1R21AG042761-01 and 1U01HL101066-01.

References

1. Lauritzen, SL. Graphical Models. Oxford, UK: Clarendon Press; 1996.
2. Arnold BC, Castillo E, Sarabia JM. Conditionally specified distributions: an introduction (with discussion). *Stat Sci*. 2001; 16:249–274.
3. Besag JE. Spatial interaction and the statistical analysis of lattice systems (with discussion). *J R Stat Soc Ser B*. 1974; 36:192–236.
4. Tierney L. Markov chains for exploring posterior distributions. *Ann Stat*. 1994; 22:1701–1728.
5. Heckerman D, Chickering DM, Meek C, Rounthwaite R, Kadie C. Dependence networks for inference, collaborative filtering, and data visualization. *J Mach Learn Res*. 2000; 1:49–57.
6. Arnold BC, Press SJ. Compatible conditional distributions. *J Am Stat Assoc*. 1989; 84:152–156.
7. Gelman A, Speed TP. Characterizing a joint probability distribution by conditionals. *J R Stat Soc Ser B*. 1993; 55:185–188.
8. Hobert JP, Casella G. Functional compatibility, Markov chains and Gibbs sampling with improper posteriors. *J Comput Graph Stat*. 1998; 7:42–60.
9. Robert, CP.; Casella, G. Monte Carlo Statistical Methods. 2nd. New York: Springer Science; 2004.
10. Wang YJ, Ip EH. Conditionally specified continuous distributions. *Biometrika*. 2008; 95:735–746.
11. Tian G, Tan M, Ng KW, Tang M. A unified method for checking compatibility and uniqueness for finite discrete conditional distributions. *Commun Stat-Theory Methods*. 2009; 38:115–129.
12. Ip EH, Wang YJ. Canonical representation of conditionally specified multivariate discrete distributions. *J Multivariate Anal*. 2009; 100:1282–1290.
13. Kuo K-L, Wang YJ. A simple algorithm for checking compatibility among discrete conditional distributions. *Comput Stat Data Anal*. 2011; 55:2457–2462.
14. Arnold BC, Gokhale DV. Distributions most nearly compatible with given families of conditional distributions. *Test*. 1998; 7:377–390.
15. Arnold BC, Castillo E, Sarabia JM. Exact and near compatibility of discrete conditional distributions. *Comput Stat Data Anal*. 2002; 16:231–252.
16. Chen S-H, Ip EH, Wang YJ. Gibbs ensembles for nearly compatible and incompatible conditional models. *Comput Stat Data Anal*. 2011; 55:1760–1769. [PubMed: 21286232]
17. Schapire RE. The strength of weak learnability. *Mach Learn*. 1990; 5:197–227.
18. Breiman L. Bagging predictors. *Mach Learn*. 1996; 24:123–140.
19. Geman S, Geman D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans Pattern Anal Mach Intell*. 1984; 6:721–741. [PubMed: 22499653]
20. Israel, RB. Gibbs distributions II. In: Kotz, S.; Read, C.; Balakrishnan, N.; Vidakovic, B., editors. *Encyclopedia of Statistical Sciences*. 2nd. Vol. 4. New York: John Wiley & Sons, Inc; 2005. p. 2838–2843.
21. Bishop, YMM.; Fienberg, SE.; Holland, PW. *Discrete Multivariate Analysis*. Cambridge, MA: MIT Press; 1975.
22. R Core Team. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing; 2012. <http://www.R-project.org/> (Accessed August 8, 2013)
23. Wand, M. KernSmooth: Functions for kernel smoothing for Wand and Jones (1995); R package version 2. 2011. p. 23–27. Available at: <http://CRAN.R-project.org/package=KernSmooth>. (Accessed August 8, 2013)
24. Rijmen F, Ip EH, Rapp S, Shaw EG. Qualitative longitudinal analysis of symptoms in patients with primary and metastatic brain tumour. *J R Stat Soc Ser A*. 2008; 171:739–753.
25. Cella DF, Tulsky DS, Gray G, Sarafian B, Linn E, Bonomi A, Siberman M, Yellen SB, Winicour P, Brannon J, et al. The functional assessment of cancer therapy scale: development and validation of the general measure. *J Clin Oncol*. 1993; 11:570–579. [PubMed: 8445433]
26. Weitzner MA, Meyers CA, Gelke CK, Byrne KS, Cella DF, Levin VA. The functional assessment of cancer therapy (fact) scale: Development of a brain subscale and revalidation of the general version (fact-g) in patients with primary brain tumors. *Cancer*. 1995; 75:1151–1161. [PubMed: 7850714]

27. Dobra A, Hans C, Jones B, Nevins JR, Yao G, West M. Sparse graphical models for exploring gene expression data. *J Multivariate Anal.* 2004; 90:196–212.

FURTHER READING

28. Arnold, BC.; Castillo, E.; Sarabia, JM. Conditionally Specification of Statistical Models. New York: Springer; 1999.
29. Bühlmann, P. Bagging, boosting and ensemble methods. In: Gentle, J.; Härdle, W.; Mori, Y., editors. *Handbook of Computational Statistics: Concepts and Methods*. Berlin, Heidelberg: Springer-Verlag; 2004. p. 877-907.
30. Casella G, George E. Explaining the Gibbs sampler. *Am Stat.* 1992; 46:167–174.
31. Levine R, Casella G. Optimizing random scan Gibbs samplers. *J Multivariate Anal.* 2006; 97:2071–2100.

FIGURE 1.

(a) Bayesian network; (b) dependency network; and (c) Markov network.

FIGURE 2.
Flow diagram for the Gibbs ensemble.

FIGURE 3.
Contour plot of joint distributions for continuous example.

TABLE 1

Equivalent Modeling Approaches with Different Terminologies

Graphical Model	Machine Learning	Reduced Distribution Representation
Acyclic graph Figure 1(a)	Bayesian network	Serial product of conditional densities and marginal density
Cyclic graph Figure 1(b)	Dependency network	CSD: A collection of conditional distributions
Undirected graph Figure 1(c)	Markov network (Markov random field)	Product of potential functions

TABLE 2
Divergence of the Joint Distributions Computed via Different Algorithms for Example 1

Errors	$\hat{\pi}^{(1,2)}$	$\hat{\pi}^{(2,1)}$	$\hat{\pi}_{LP}$	$\hat{\pi}_{RS}$	$\hat{\pi}_{EQ}$	$\hat{\pi}_L$	$\hat{\pi}_I$	$\hat{\pi}_X$	$\hat{\pi}_F$
L^2	0.094	0.278	0.091	0.094	0.093	0.070	0.071	0.077	0.072
$\hat{P}$	0.139	0.311	0.113	0.117	0.115	0.097	0.096	0.101	0.097
χ^2	0.383	0.594	0.209	0.214	0.212	0.214	0.202	0.198	0.200
F^2	0.292	0.592	0.220	0.226	0.223	0.197	0.193	0.198	0.193

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

TABLE 3

Comparisons of Divergence for the Continuous Example

	L^2			F^2		
	p_1	p_2	Total	p_1	p_2	Total
$\hat{\pi}^{(1,2)}$	2.6269	0.6620	3.2889	2587.95	409.42	2997.37
$\hat{\pi}^{(2,1)}$	1.2112	1.9534	3.1646	1089.14	1312.94	2402.08
$\tilde{\pi}_{EQ}$	1.7639	0.9586	2.7225	1627.03	530.00	2157.03
$\tilde{\pi}_L$	1.7521	0.9722	2.7243	1614.80	536.12	2150.92
$\tilde{\pi}_F$	1.6982	1.0390	2.7372	1558.46	566.73	2125.19

Divergences of 10 Randomly Selected Pseudo-Gibbs Solutions (PGS1–PGS10) and the Corresponding s -Nested Gibbs Ensembles (s_1 – s_{10})**TABLE 4**

	PGS1	PGS2	PGS3	PGS4	PGS5	PGS6	PGS7	PGS8	PGS9	PGS10
L^2	0.710	1.547	0.248	0.583	0.703	0.675	1.210	1.565	0.409	1.264
I^2	2.300	4.662	0.738	1.630	2.333	2.220	3.836	4.716	1.187	4.029
X^2	1.975	5.055	0.719	1.680	2.039	1.875	4.234	5.123	1.094	4.098
s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	
L^2	0.248	0.216	0.159	0.149	0.152	0.163	0.178	0.181	0.199	0.219
I^2	0.738	0.650	0.492	0.504	0.545	0.596	0.650	0.678	0.739	0.804
X^2	0.719	0.597	0.462	0.452	0.473	0.522	0.530	0.571	0.616	0.664