

On advancing MCMC-based methods for Markovian data structures with applications to deep learning, simulation, and resampling

Andee Kaplan

Iowa State University
ajkaplan@iastate.edu

February 10, 2017

Slides available at <http://bit.ly/kaplan-cornell>

Joint work with D. Nordman and S. Vardeman

Overview

Goal: Develop statistical inference via Markov chain Monte Carlo (MCMC) techniques in complex data problems related to statistical learning, the analysis of network/graph data, and spatial resampling.

Challenge: Develop implementations which are both *statistically rigorous* and *computationally scalable* by exploiting conditional independence, or Markov structures, in data models.

- 1 Statistical quantification of graph models used in deep machine learning and image classification
- 2 New, fast methods for simulating spatial, network, and other data with complex dependencies
- 3 Developments in interactive graphics and statistical applications for the web

On the propriety of restricted Boltzmann machines

What is this?

A restricted Boltzman machine (RBM) is an undirected probabilistic graphical model with

- 1 two layers of random variables - one hidden and one visible
- 2 conditional independence within a layer (Smolensky 1986)

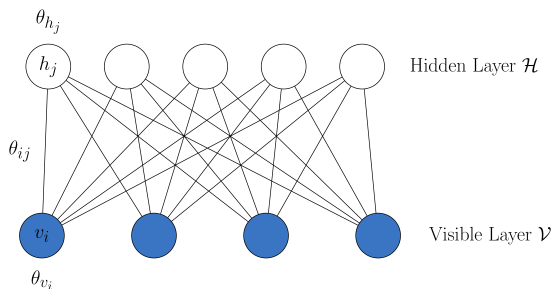


Figure 1: Hidden nodes are indicated by white circles and the visible nodes are indicated by blue circles.

How is it used?

- Supervised learning, specifically image classification

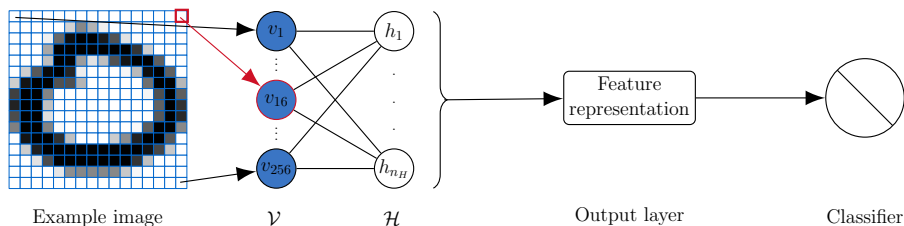


Figure 2: Image classification using a RBM: each image pixel comprises a node in the visible layer, \mathcal{V} and the output of the RBM is used to create features passed to a supervised learning algorithm.

Joint distribution

- $\mathbf{x} = (h_1, \dots, h_{n_H}, v_1, \dots, v_{n_V})$ represents visible and hidden nodes in a RBM
- Each single “binary” random variable, visible v_i or hidden h_j , takes values in a common coding set
 - $\mathcal{C} = \{0, 1\}$ or $\mathcal{C} = \{-1, 1\}$.
- A parametric form for probabilities

$$f_{\theta}(\mathbf{x}) = \frac{\exp \left(\sum_{i=1}^{n_V} \sum_{j=1}^{n_H} \theta_{ij} v_i h_j + \sum_{i=1}^{n_V} \theta_{v_i} v_i + \sum_{j=1}^{n_H} \theta_{h_j} h_j \right)}{\gamma(\theta)}$$

where

$$\gamma(\theta) = \sum_{\mathbf{x} \in \mathcal{C}^{n_H+n_V}} \exp \left(\sum_{i=1}^{n_V} \sum_{j=1}^{n_H} \theta_{ij} v_i h_j + \sum_{i=1}^{n_V} \theta_{v_i} v_i + \sum_{j=1}^{n_H} \theta_{h_j} h_j \right)$$

Deep learning

- Stacking layers of RBMs in a deep architecture
- Proponents claim the ability to learn "internal representations that become increasingly complex, which is considered to be a promising way of solving object and speech recognition problems" (Salakhutdinov and Hinton 2009, pp. 450).

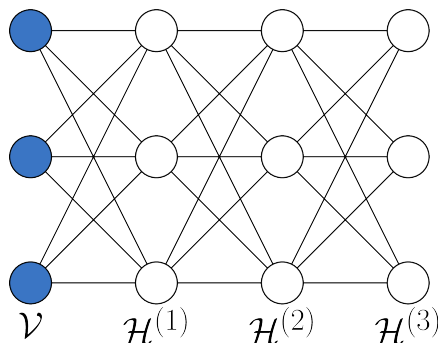


Figure 3: Three layer deep Boltzmann machine, with visible-to-hidden and hidden-to-hidden connections but no within-layer connections.

Why do I care?

- ① The model properties are largely unexplored in the literature
- ② The commonly cited fitting methodology remains heuristic-based and abstruse (Hinton, Osindero, and Teh 2006)

We want to

- ① Provide steps toward understanding properties of the model class from the perspective of statistical theory
- ② Explore the possibility of a rigorous fitting methodology

Degeneracy, instability, and uninterpretability. Oh my!

The highly flexible nature of a RBM ($n_H + n_V + n_H * n_V$ parameters) makes at least three kinds of potential model impropriety of concern

- ① *degeneracy*
- ② *instability*, and
- ③ *uninterpretability*

A model should “provide an explanation of the mechanism underlying the observed phenomena” (G. E. P. Box 1967).

RBM often

- fail to generate data with realistic variability and thus an unsatisfactory conceptualization of the data generation process (Li 2014)
- exhibit model instability (over-sensitivity) (Szegedy et al. 2013; Nguyen, Yosinski, and Clune 2014)

Near-degeneracy

Definition (Model Degeneracy)

A disproportionate amount of probability is placed on only a few elements of the sample space, $\mathcal{C}^{n_H+n_V}$, by the model.

RBM models exhibit *near-degeneracy* when random variables in

$$Q_{\theta}(\mathbf{x}) = \sum_{i=1}^{n_V} \sum_{j=1}^{n_H} \theta_{ij} v_i h_j + \sum_{i=1}^{n_V} \theta_{v_i} v_i + \sum_{j=1}^{n_H} \theta_{h_j} h_j,$$

have a mean vector $\boldsymbol{\mu}(\boldsymbol{\theta})$ close to the boundary of the convex hull of $\mathcal{T} = \{\mathbf{t}(\mathbf{x}) : \mathbf{x} \in \mathcal{C}^{n_H+n_V}\}$ (Handcock 2003), where

$$\mathbf{t}(\mathbf{x}) = \{v_1, \dots, v_{n_V}, h_1, \dots, h_{n_H}, v_1 h_1, \dots, v_{n_V} h_{n_H}\}$$

and

$$\boldsymbol{\mu}(\boldsymbol{\theta}) = \mathbb{E}_{\theta} \mathbf{t}(\mathbf{X})$$

Instability

Definition (Instability)

Characterized by excessive sensitivity in the model, where small changes in the components of data outcomes, \mathbf{x} , lead to substantial changes in probability.

- Concept of model deficiency related to *instability* for a class of exponential families of distributions (Schweinberger 2011)
- For the RBM, consider how model incorporates more visibles
 - Model parameters in a longer sequence $\theta_{n_V} \in \mathbb{R}^{n_V + n_H + n_V * n_H}$, $n_V \geq 1$
 - May also arbitrarily expand the number of hidden variables used

Unstable RBMs

Definition (S-unstable RBM)

A RBM model formulation is *S-unstable* if

$$\lim_{n_V \rightarrow \infty} \frac{1}{n_V} \text{ELPR}(\theta_{n_V}) = \infty.$$

where

$$\text{ELPR}(\theta_{n_V}) = \log \left[\frac{\max_{(v_1, \dots, v_{n_V}) \in \mathcal{C}^{n_V}} P_{\theta_{n_V}}(v_1, \dots, v_{n_V})}{\min_{(v_1, \dots, v_{n_V}) \in \mathcal{C}^{n_V}} P_{\theta_{n_V}}(v_1, \dots, v_{n_V})} \right] \quad (1)$$

S-unstable RBM models are undesirable for several reasons - small changes in data outcomes can lead to overly-sensitive changes in probability.

One-pixel change

Consider the biggest log-probability ratio for a one-pixel (one component) change in visibles (data outcomes)

$$\Delta(\theta_{n_V}) \equiv \max \left\{ \log \frac{P_{\theta_{n_V}}(v_1, \dots, v_{n_V})}{P_{\theta_{n_V}}(v_1^*, \dots, v_{n_V}^*)} \right\},$$

where (v_1, \dots, v_{n_V}) & $(v_1^*, \dots, v_{n_V}^*) \in \mathcal{C}^{n_V}$ differ by exactly one component

Result

Let $c > 0$ and fix an integer $n_V \geq 1$. If $\frac{1}{n_V} ELPR(\theta_{n_V}) > c$, then $\Delta(\theta_{n_V}) > c$.

If the $n_V^{-1} ELPR(\theta_{n_V})$ is too large, then a RBM model will exhibit large probability shifts for very small changes in the data configuration.

Tie to degeneracy

Define an arbitrary modal set of possible outcomes (i.e. set of highest probability outcomes) for a given $0 < \epsilon < 1$ as

$$M_{\epsilon, \theta_{n_V}} \equiv \left\{ \mathbf{v} \in \mathcal{C}^{n_V} : \log P_{\theta_{n_V}}(\mathbf{v}) > (1 - \epsilon) \max_{\mathbf{v}^*} P_{\theta_{n_V}}(\mathbf{v}^*) + \epsilon \min_{\mathbf{v}^*} P_{\theta_{n_V}}(\mathbf{v}^*) \right\}$$

Result

For an S-unstable RBM model, and for any given $0 < \epsilon < 1$, $P_{\theta_{n_V}}((v_1, \dots, v_{n_V}) \in M_{\epsilon, \theta_{n_V}}) \rightarrow 1$ holds as $n_V \rightarrow \infty$.

- All probability will stack up on mode sets or potentially those few outcomes with the highest probability
- Proofs found in (A. Kaplan, Nordman, and Vardeman 2016)

Uninterpretability

Definition (Uninterpretability)

Characterized by marginal mean-structure (controlled by main effect parameters $\theta_{v_i}, \theta_{h_j}$) not being maintained in the model due to dependence (interaction parameters θ_{ij}) (Kaiser 2007).

- Model expectations, $E[\mathbf{X}|\boldsymbol{\theta}]$
- Expectations given independence, $E[\mathbf{X}|\boldsymbol{\theta}^*]$, where $\boldsymbol{\theta}^*$ matches $\boldsymbol{\theta}$ for all main effects but otherwise has $\theta_{ij} = 0$ for $i = 1, \dots, n_V, j = 1, \dots, n_H$
- If $|E[\mathbf{X}|\boldsymbol{\theta}] - E[\mathbf{X}|\boldsymbol{\theta}^*]|$ is large then the RBM with parameter vector $\boldsymbol{\theta}$ is *uninterpretable*

Manageable (a.k.a. small) examples

- To explore the effects of RBM parameters θ on *near-degeneracy*, *instability*, and *uninterpretability*, consider models of small size
- For $n_H, n_V \in \{1, \dots, 4\}$, sample 100 values of θ
 - 1 Split θ into $\theta_{interaction}$ and θ_{main}
 - 2 Allow the two types of terms to have varying average magnitudes, $\|\theta_{main}\|/(n_H + n_V)$ and $\|\theta_{interaction}\|/(n_H * n_V)$
 - 3 Average magnitudes vary on a grid between 0.001 and 3 with 24 breaks, yielding 576 grid points
- Calculate metrics of model impropriety, $\mu(\theta)$, $ELPR(\theta)/n_V$, and the coordinates of $|E[X|\theta] - E[X|\theta^*]|$.
- In the case of *near-degeneracy*, classify each model as near-degenerate or “viable” based on the distance of $\mu(\theta)$ from the boundary of the convex hull of \mathcal{T}

Simulation results

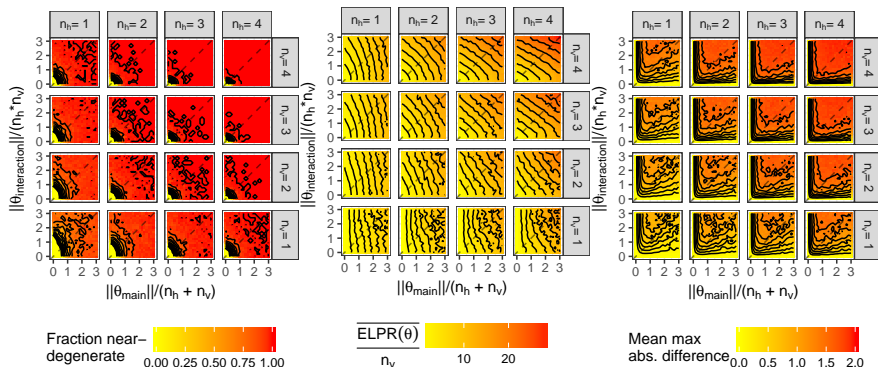


Figure 4: The fraction of models that were near-degenerate (left), the sample mean value of $\text{ELPR}(\theta)/n_v$ (middle), and the sample mean of the maximum component of the absolute difference between the model expectation vector, $E[\mathbf{X}|\theta]$, and the expectation vector given independence, $E[\mathbf{X}|\theta^*]$ (right).

Model fitting

- ① Computational concerns: Fitting a RBM via maximum likelihood (ML) methods infeasible due to the intractability of the normalizing term $\gamma(\theta)$
 - Ad hoc methods used to avoid this problem with stochastic ML
 - Employ a small number of MCMC draws to approximate $\gamma(\theta)$
- ② Model parameterization concerns: With enough hidden units,
 - Potential to re-create any distribution for the data (Le Roux and Bengio 2008; Montufar and Ay 2011; and Montúfar, Rauh, and Ay 2011)
 - The model for the cell probabilities that has the highest likelihood over *all possible model classes* is the empirical distribution
 - The RBM model ensures that this empirical distribution can be arbitrarily well approximated
 - When empirical distribution contains empty cells, ML will chase parameters to ∞ in order to zero out corresponding RBM cell probabilities

Bayesian methods

- Consider what might be done in a principled manner, small test
- To avoid model impropriety, avoid parts of the parameter space $\mathbb{R}^{n_V+n_H+n_V*n_H}$ leading to *near-degeneracy*, *instability*, and *uninterpretability*.
 - Shrink θ toward $\mathbf{0}$
 - 1 Specify priors that place low probability on large values of $||\theta||$
 - 2 Shrink $\theta_{interaction}$ more than θ_{main}
- Consider a test case with $n_V = n_H = 4$
 - θ chosen as a sampled value from a grid point in figure 4 with $< 5\%$ near-degeneracy (not near the convex hull of the sufficient statistics)
 - simulate $n = 5,000$ as a training set and fit the RBM using three Bayes methodologies

Fitting methodologies

① A “trick” prior (BwTPLV)

- Cancel out normalizing term in the likelihood
- Resulting full conditionals of θ are multivariate Normal
- h_j are carried along as latent variables

$$\pi(\theta) \propto \gamma(\theta)^n \exp \left(-\frac{1}{2C_1} \theta'_{main} \theta_{main} - \frac{1}{2C_2} \theta'_{interaction} \theta_{interaction} \right),$$

where $C_2 < C_1$ (Li 2014)

Fitting methodologies (cont'd)

② *A truncated Normal prior (BwTNLV)*

- Independent spherical normal distributions as priors for θ_{main} and $\theta_{interaction}$
 - $\sigma_{interaction} < \sigma_{main}$
 - *truncated* at $3\sigma_{main}$ and $3\sigma_{interaction}$, respectively
- Simulation from the posterior using a geometric adaptive MH step (Zhou 2014)
- h_j are carried along in the MCMC implementation as latent variables

③ *A truncated Normal prior and marginalized likelihood (BwTNML)*

- Marginalize out \mathbf{h} in $f_{\theta}(\mathbf{x})$
- Use the truncated Normal priors applied to the marginal probabilities for visible variables (recall visibles are the observed data, hiddens are not)

Mixing

The BwTNLV (2) and the BwTNML method (3) are drawing from the same stationary posterior distribution for images.

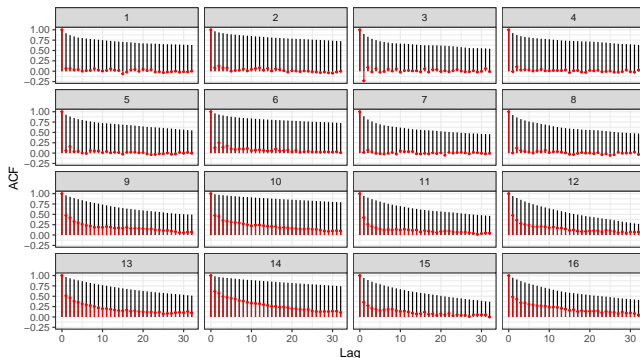


Figure 5: The autocorrelation functions (ACF) for the posterior probabilities of all $2^4 = 16$ possible outcomes for the vector of 4 visibles assessed at multiple lags for each method with BwTNLV in black and BwTNML in red.

Posterior distributions of images

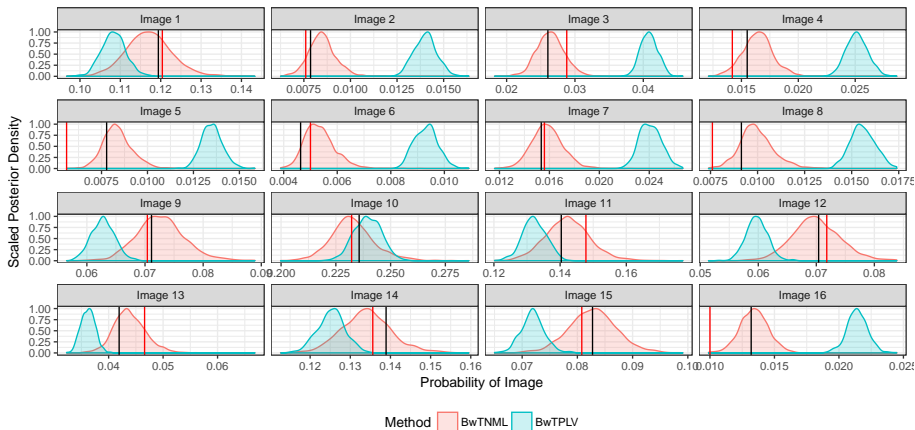


Figure 6: Posterior probabilities of $16 = 2^4$ possible realizations of 4 visibles using two of the three Bayesian fitting techniques, BwTPLV and BwTNML. Black lines show true probabilities of each vector of visibles based on the parameters used to generate the training data while red lines show the empirical distribution.

Wrapping up

- RBMs used for classification, but are concerning as statistical models due to *near-degeneracy*, *S-instability*, and *uninterpretability*
- Rigorous fitting methodology is difficult due to the dimension of the parameter space & size of the latent variable space
- Fitting a RBM model is also questionable as any distribution for the visibles can be approximated arbitrarily well
 - The empirical distribution of visibles is the best fitting model for observed cell data
 - There can be no “smoothed distribution” achieved in a RBM model of sufficient size with a rigorous likelihood-based method

Skeptical that any model built using RBMs (i.e. deep Boltzmann machine) can achieve useful **prediction** or **inference** in a principled way without limiting the flexibility of the fitted model

A simple, fast sampler for simulating spatial data and other Markovian data structures

Goal

- Markov random field models are possible for spatial or network data
- Rather than specifying a joint distribution directly, a model is specified through a set of full conditional distributions for each spatial location
- Assume the spatial data are on a regular lattice (wrapped on a torus)

Goal: A new, provably fast approach for simulating spatial/network data.

Spatial Markov random field (MRF) models

Notation

- Variables $\{Y(\mathbf{s}_i) : i = 1, \dots, n\}$ at locations $\{\mathbf{s}_i : i = 1, \dots, n\}$
- Neighborhoods: N_i specified according to some configuration
- Neighboring Values: $\mathbf{y}(N_i) = \{y(\mathbf{s}_j) : \mathbf{s}_j \in N_i\}$
- Full Conditionals: $\{f_i(y(\mathbf{s}_i) | \mathbf{y}(N_i), \boldsymbol{\theta}) : i = 1, \dots, n\}$
 - $f_i(y(\mathbf{s}_i) | \mathbf{y}(N_i), \boldsymbol{\theta})$ is conditional pmf/pdf of $Y(\mathbf{s}_i)$ given values for its neighbors $\mathbf{y}(N_i)$
 - Often assume a common conditional cdf $F_i = F$ form ($f_i = f$) for all i

Exponential family examples

- 1 Conditional Gaussian (3 parameters):

$$f_i(y(\mathbf{s}_i) | \mathbf{y}(N_i), \alpha, \eta, \tau) = \frac{1}{\sqrt{2\pi\tau}} \exp\left(-\frac{[y(\mathbf{s}_i) - \mu(\mathbf{s}_i)]^2}{2\tau^2}\right)$$

$Y(\mathbf{s}_i)$ given neighbors $\mathbf{y}(N_i)$ is normal with variance τ^2 and mean

$$\mu(\mathbf{s}_i) = \alpha + \eta \sum_{\mathbf{s}_j \in N_i} [y(\mathbf{s}_j) - \alpha]$$

- 2 Conditional Binary (2 parameters):

$Y(\mathbf{s}_i)$ given neighbors $\mathbf{y}(N_i)$ is Bernoulli $p(\mathbf{s}_i, \kappa, \eta)$ where

$$\text{logit}[p(\mathbf{s}_i, \kappa, \eta)] = \text{logit}(\kappa) + \eta \sum_{\mathbf{s}_j \in N_i} [y(\mathbf{s}_j) - \kappa]$$

In both examples, η represents a dependence parameter.

Concliques

Cliques – Hammersley and Clifford (1971)

Singletons and sets of locations such that each location in the set is a neighbor of all other locations in the set

Example: Four nearest neighbors gives cliques of sizes 1 and 2

The Converse of Cliques – Concliques

Sets of locations such that no location in the set is a neighbor of any other location in the set

4 Nearest
Neighbors

.	*	.
*	S	*
.	*	.

Concliques
4 Nearest
Neighbors

1	2	1	2
2	1	2	1
1	2	1	2
2	1	2	1

8 Nearest
Neighbors

*	*	*
*	S	*
*	*	*

Concliques
8 Nearest
Neighbors

1	2	1	2
3	4	3	4
1	2	1	2
3	4	3	4

Generalized spatial residuals

Definition

- $F(y|\mathbf{y}(N_i), \boldsymbol{\theta})$ is the conditional cdf of $Y(\mathbf{s}_i)$ under the model
- Substitute random variables, $Y(\mathbf{s}_i)$ and neighbors $\{Y(\mathbf{s}_j) : \mathbf{s}_j \in N_i\}$, into (continuous) conditional cdf to define residuals:

$$R(\mathbf{s}_i) = F(Y(\mathbf{s}_i) | \{Y(\mathbf{s}_j) : \mathbf{s}_j \in N_i\}, \boldsymbol{\theta}).$$

Key Property

Let $\{\mathcal{C}_j : j = 1, \dots, q\}$ be a collection of cliques that partition the integer grid. Under the conditional model, **spatial residuals within a clique are iid Uniform(0, 1)-distributed**:

$$\{R(\mathbf{s}_i) : \mathbf{s}_i \in \mathcal{C}_j\} \stackrel{iid}{\sim} \text{Uniform}(0, 1) \quad \text{for } j = 1, \dots, q$$

(M. S. Kaiser, Lahiri, and Nordman 2012)

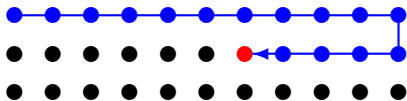
Common Spatial Simulation Approach

With common conditionally specified models for spatial lattice, standard MCMC simulation approach via Gibbs sampling is:

Starting from some initial $\mathbf{Y}_*^{(j)} \equiv \{Y_*^{(j)}(\mathbf{s}_1), \dots, Y_*^{(j)}(\mathbf{s}_n)\}$,

- 1 Moving row-wise, for $i = 1, \dots, n$, individually simulate/update $Y_*^{(j+1)}(\mathbf{s}_i)$ for each location \mathbf{s}_i from conditional cdf F given

$$Y_*^{(j+1)}(\mathbf{s}_1), \dots, Y_*^{(j+1)}(\mathbf{s}_{i-1}), \quad Y_*^{(j)}(\mathbf{s}_{i+1}), \dots, Y_*^{(j)}(\mathbf{s}_n)$$



- 2 n individual updates provide 1 full Gibbs iteration.
- 3 Repeat 1-2 to obtain M resampled spatial data sets $\mathbf{Y}_*^{(j)}$, $j = 1, \dots, M$ (e.g., can burn-in, thin, etc.)

Conclique-based Gibbs sampler

Using the conditional independence of random variables at locations within a conclique along with the probability integral transform we propose a conclique-based Gibbs sampling algorithm for sampling from a MRF.

- ① Split locations into Q disjoint concliques, $\mathcal{D} = \cup_{i=1}^Q \mathcal{C}_i$.
- ② Initialize the values of $\{Y^{(0)}(\mathbf{s}) : \mathbf{s} \in \{\mathcal{C}_2, \dots, \mathcal{C}_Q\}\}$.
- ③ Sample from the conditional distribution of $Y(\mathbf{s})$ given $\{Y(\mathbf{t}) : \mathbf{t} \in \mathcal{N}(\mathbf{s})\}$ for $\mathbf{s} \in \mathcal{C}_1$,
 - ① Sample $\{U(s) : s \in \mathcal{C}_1\} \stackrel{iid}{\sim} Unif(0, 1)$
 - ② For each $\mathbf{s} \in \mathcal{C}_1$, $Y^{(i)}(\mathbf{s}) = F^{-1}(U(\mathbf{s}) | Y^{(i-1)}(\mathbf{t}), \mathbf{t} \in \mathcal{N}(\mathbf{s}))$
- ④ Sample from the conditional distribution of $Y(\mathbf{s})$ given $\{Y(\mathbf{t}) : \mathbf{t} \in \mathcal{N}(\mathbf{s})\}$ for $\mathbf{s} \in \mathcal{C}_j; j = 2, \dots, Q$,
 - ① Sample $\{U(s) : s \in \mathcal{C}_j\} \stackrel{iid}{\sim} Unif(0, 1)$
 - ② For each $\mathbf{s} \in \mathcal{C}_j$, $Y^{(i)}(\mathbf{s}) = F^{-1}(U(\mathbf{s}) | \{Y^{(i)}(\mathbf{t}), \mathbf{t} \in \mathcal{N}(\mathbf{s}) \cap \mathcal{C}_k \text{ where } k < j\}, \{Y^{(i-1)}(\mathbf{t}), \mathbf{t} \in \mathcal{N}(\mathbf{s}) \cap \mathcal{C}_k \text{ where } k > j\})$

It's (provably) fast!

- ① In many (commonly used) four-nearest neighbor models (including Gaussian and binary), the conclique-based Gibbs sampler is provably **geometrically ergodic**.
- ② Because we are using batch updating vs. sequential updating of each location, this approach is also **computationally fast**.
- ③ A flexible R package using Rcpp (called conclique, to appear on CRAN) that implements a conclique-based Gibbs sampler while allowing the user to specify an arbitrary model.

Preliminary simulations

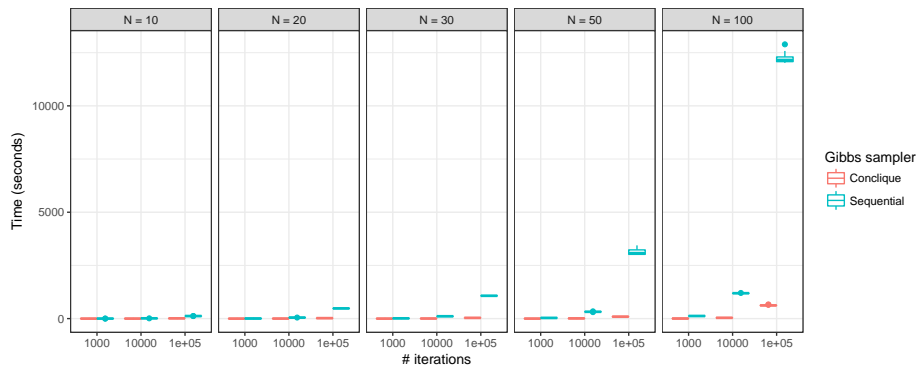


Figure 7: Comparisons of timing for simulation of 4NN Gaussian Markov Random Field data on a lattice of size $N \times N$ for various size grids, $N = 10, 20, 30, 50, 100$, using sequential and conclave-based Gibbs samplers.

Other projects

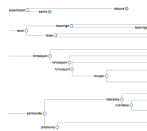
Statistical software for teaching

intRo (<http://intro-stats.com>): a web application for performing basic data analyses and statistical routines in the classroom (Hare and Kaplan 2017)

- A simple web-based application for performing basic data analysis and statistical routines and accompanying utility package
- Built using R and Shiny
- Extensible modular structure
- Designed for a first statistics class student
- Assists in the learning of statistics rather than acting as a stand-alone deliverer of statistics education

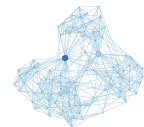
Ultterior motive: get students excited about programming

Interactive web graphics

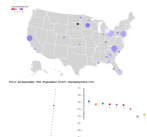


Protoshiny Application to interactively visualize hierarchical clustering with prototypes through expansion and pan/zoom of dendrogram.

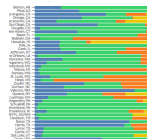
gravicom Graphical Visualization of Communities. A web application for community detection in network data through direct user interaction.



CommuniD3 Exploratory tool based on 'Soul of the Community' data generated by the Knight Foundation in cooperation with Gallup. 1st place 2013 Data Exposition.



NCS Dataviz A graphical tool that allows users to understand variable relationships within the 2012 NCS Vanguard Study dataset.



Future plans

Ideas and connections

- ① Generalization of instability results for other network models (ongoing, see A. Kaplan, Nordman, and Vardeman 2016)
- ② Image classification
 - Ensemble methods (super learners) using AdaBoost (Freund and Schapire 1995)
 - Decision theoretic approach to approximating the LRT for classification
- ③ Resampling methods for network data
 - Parametric bootstrap (GOF test)
 - Model-based method of resampling re-frames network into a collection of neighborhoods by using covariate information (concliques)
 - Conditionally specified network distribution (Casleton, Nordman, and Kaiser 2016) to sample network data in a conclique-based Gibbs sampler.
 - Nonparametric bootstrap
 - Does not impose a neighborhood structure, nor distributional assumptions
 - Analogous to the block bootstrap in time series - network data blocks created by a local topological rule

Thank you

- Slides – <http://bit.ly/kaplan-cornell>
- Contact
 - Email – ajkaplan@iastate.edu
 - Twitter – <http://twitter.com/andeekaplan>
 - GitHub – <http://github.com/andeek>

Appendix

Data coding to mitigate degeneracy

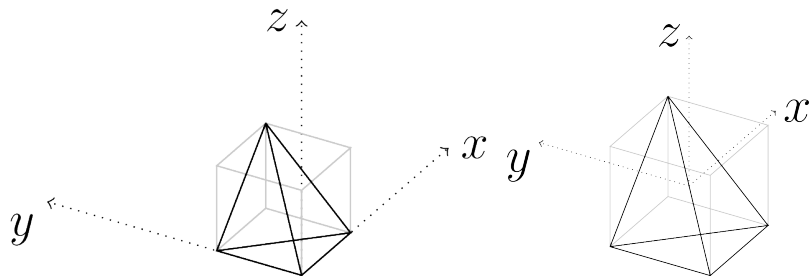


Figure 8: The convex hull of the "statistic space" $\mathcal{T} = \{(v_1, h_1, v_1 h_1) : v_1, h_1 \in \mathcal{C}\}$ in three dimensions for a toy RBM with one visible and one hidden node for $\mathcal{C} = \{0, 1\}$ (left) and $\mathcal{C} = \{-1, 1\}$ (right) data encoding.

The convex hull of $\mathcal{T} \subset \mathcal{C}^3$ does not fill the unit cube $[0, 1]^3$ (left), but does better with $[-1, 1]^3$ (right).

The center of the universe

- For the $\mathcal{C} = \{-1, 1\}$ encoding of hidden (H_1, \dots, H_{n_H}) and visible (V_1, \dots, V_{n_V}) , the origin is the center of the parameter space.
- At $\theta = \mathbf{0}$, RBM is equivalent to elements of X being distributed as iid Bernoulli $\left(\frac{1}{2}\right) \Rightarrow$ No *near-degeneracy*, *instability*, or *uninterpretability*!

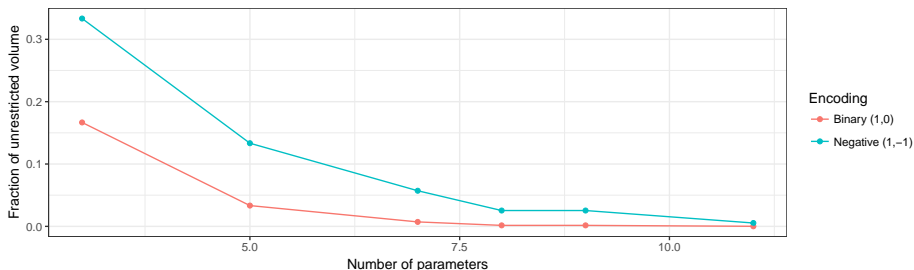


Figure 9: Relationship between volume of the convex hull of possible values of the RBM sufficient statistics and the cube containing it for different size models.

Parameters used

Table 1: Parameters used to fit a test case with $n_V = n_H = 4$. This parameter vector was chosen as a sampled value of θ that was not near the convex hull of the sufficient statistics for a grid point in figure 4 with $< 5\%$ near-degeneracy.

Parameter	Value	Parameter	Value	Parameter	Value
θ_{v1}	-1.1043760	θ_{11}	-0.0006334	θ_{31}	-0.0038301
θ_{v2}	-0.2630044	θ_{12}	-0.0021401	θ_{32}	0.0032237
θ_{v3}	0.3411915	θ_{13}	0.0047799	θ_{33}	0.0020681
θ_{v4}	-0.2583769	θ_{14}	0.0025282	θ_{34}	0.0041429
θ_{h1}	-0.1939302	θ_{21}	0.0012975	θ_{41}	0.0089533
θ_{h2}	-0.0572858	θ_{22}	0.0000253	θ_{42}	-0.0042403
θ_{h3}	-0.2101802	θ_{23}	-0.0004352	θ_{43}	-0.0000480
θ_{h4}	0.2402456	θ_{24}	-0.0086621	θ_{44}	0.0004767

Effective sample size

- Overlapping blockmeans approach (Gelman, Shirley, and others 2011)
 - Crude estimate for the asymptotic variance of the probability of each image
 - Compare it to an estimate of the asymptotic variance assuming IID draws from the target distribution

Table 2: The effective sample sizes for a chain of length $M = 1000$ regarding all 16 probabilities for possible vector outcomes of visibles. BwTNLV would require at least 4.7 times as many MCMC iterations to achieve the same amount of effective information about the posterior distribution.

Outcome	BwTNLV	BwTNML	Outcome	BwTNLV	BwTNML
1	73.00	509.43	9	83.47	394.90
2	65.05	472.51	10	95.39	327.35
3	87.10	1229.39	11	70.74	356.56
4	72.64	577.73	12	81.40	338.30
5	71.67	452.01	13	105.98	373.59
6	66.49	389.78	14	132.61	306.91
7	84.30	660.37	15	82.15	365.30
8	75.46	515.09	16	98.05	304.57

References I

- Casleton, Emily, Daniel Nordman, and Mark Kaiser. 2016. "A Local Structure Model for Network Analysis." *Statistics and Its Interface, Forthcoming*.
- Freund, Yoav, and Robert E Schapire. 1995. "A Desicion-Theoretic Generalization of on-Line Learning and an Application to Boosting." In *European Conference on Computational Learning Theory*, 23–37. Springer.
- G. E. P. Box, W. J. Hill. 1967. "Discrimination Among Mechanistic Models." *Technometrics* 9 (1). [Taylor & Francis, Ltd., American Statistical Association, American Society for Quality]: 57–71.
- Gelman, Andrew, Kenneth Shirley, and others. 2011. "Inference from Simulations and Monitoring Convergence." *Handbook of Markov Chain Monte Carlo*, 163–74.
- Hammersley, John M, and Peter Clifford. 1971. "Markov Fields on Finite Graphs and Lattices."
- Handcock, Mark S. 2003. "Assessing Degeneracy in Statistical Models of Social Networks." Center for Statistics; the Social Sciences, University of Washington. <http://www.csss.washington.edu/>.
- Hare, Eric, and A. Kaplan. 2017. "Designing Modular Software: A Case Study in Introductory Statistics." *Journal of Computational and Graphical Statistics* 0 (ja). <http://amstat.tandfonline.com/doi/abs/10.1080/10618600.2016.1276839>.
- Hinton, Geoffrey E, Simon Osindero, and Yee-Whye Teh. 2006. "A Fast Learning Algorithm for Deep Belief Nets." *Neural Computation* 18 (7). MIT Press: 1527–54.
- Kaiser, Mark S. 2007. "Statistical Dependence in Markov Random Field Models." *Statistics Preprints Paper* 57. Digital Repository @ Iowa State University. http://lib.dr.iastate.edu/stat_las_preprints/57/.
- Kaiser, Mark S, Soumendra N Lahiri, and Daniel J Nordman. 2012. "Goodness of Fit Tests for a Class of Markov Random Field Models." *The Annals of Statistics*. JSTOR, 104–30.
- Kaplan, Andee, Daniel Nordman, and Stephen Vardeman. 2016. "A Note on the Instability and Degeneracy of Deep Learning

References II

Models." *Under Review*.

Le Roux, Nicolas, and Yoshua Bengio. 2008. "Representational Power of Restricted Boltzmann Machines and Deep Belief Networks." *Neural Computation* 20 (6). MIT Press: 1631–49.

Li, Jing. 2014. "Biclustering Methods and a Bayesian Approach to Fitting Boltzmann Machines in Statistical Learning." PhD thesis, Iowa State University; Graduate Theses; Dissertations. <http://lib.dr.iastate.edu/etd/14173/>.

Montufar, Guido, and Nihat Ay. 2011. "Refinements of Universal Approximation Results for Deep Belief Networks and Restricted Boltzmann Machines." *Neural Computation* 23 (5). MIT Press: 1306–19.

Montúfar, Guido F, Johannes Rauh, and Nihat Ay. 2011. "Expressive Power and Approximation Errors of Restricted Boltzmann Machines." In *Advances in Neural Information Processing Systems*, 415–23. NIPS.

Nguyen, Anh Mai, Jason Yosinski, and Jeff Clune. 2014. "Deep Neural Networks Are Easily Fooled: High Confidence Predictions for Unrecognizable Images." *ArXiv Preprint ArXiv:1412.1897*. <http://arxiv.org/abs/1412.1897>.

Salakhutdinov, Ruslan, and Geoffrey E Hinton. 2009. "Deep Boltzmann Machines." In *International Conference on Artificial Intelligence and Statistics*, 448–55. AI & Statistics.

Schweinberger, Michael. 2011. "Instability, Sensitivity, and Degeneracy of Discrete Exponential Families." *Journal of the American Statistical Association* 106 (496). Taylor & Francis: 1361–70.

Smolensky, Paul. 1986. "Information Processing in Dynamical Systems: Foundations of Harmony Theory." DTIC Document.

Szegedy, Christian, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2013. "Intriguing Properties of Neural Networks." *ArXiv Preprint ArXiv:1312.6199*. <http://arxiv.org/abs/1312.6199>.

Zhou, Wen. 2014. "Some Bayesian and Multivariate Analysis Methods in Statistical Machine Learning and Applications." PhD thesis, Iowa State University; Graduate Theses; Dissertations. <http://lib.dr.iastate.edu/etd/13816/>.