Probabilistic Graphical Models

Brown University CSCI 2950-P, Spring 2013
Prof. Erik Sudderth

Lecture 17:
Collapsed Gibbs Samplers,
MCMC Mixing and Diagnostics

Some slides and figures courtesy Iain Murray’s tutorial,
Markov Chain Monte Carlo, MLSS 2009
Review: MCMC Methods

Construct a biased random walk that explores a target dist.

Markov steps, \( x^{(s)} \sim T\left(x^{(s)} \leftarrow x^{(s-1)}\right) \)

MCMC gives approximate, correlated samples

\[
\mathbb{E}_P[f] \approx \frac{1}{S} \sum_{s=1}^{S} f(x^{(s)})
\]

Example transitions:

**Metropolis–Hastings:** \( T(x' \leftarrow x) = Q(x'; x) \min\left(1, \frac{P(x') Q(x; x')}{P(x) Q(x'; x)}\right) \)

**Gibbs sampling:** \( T_i(x' \leftarrow x) = P(x'_i | x_{j \neq i}) \delta(x'_{j \neq i} - x_{j \neq i}) \)
Combining MCMC Transition Proposals

A sequence of operators, each with $P^*$ invariant:

\[
\begin{align*}
x_0 & \sim P^*(x) \\
x_1 & \sim T_a(x_1 \leftarrow x_0) \quad P(x_1) = \sum_{x_0} T_a(x_1 \leftarrow x_0)P^*(x_0) = P^*(x_1) \\
x_2 & \sim T_b(x_2 \leftarrow x_1) \quad P(x_2) = \sum_{x_1} T_b(x_2 \leftarrow x_1)P^*(x_1) = P^*(x_2) \\
x_3 & \sim T_c(x_3 \leftarrow x_2) \quad P(x_3) = \sum_{x_1} T_c(x_3 \leftarrow x_2)P^*(x_2) = P^*(x_3) \\
\end{align*}
\]

\[
\begin{align*}
&\cdots \\
\end{align*}
\]

— Combination $T_cT_bT_a$ leaves $P^*$ invariant
— If they can reach any $x$, $T_cT_bT_a$ is a valid MCMC operator
— Individually $T_c$, $T_b$ and $T_a$ need not be ergodic
Gibbs Samplers

A method with no rejections:

– Initialize \(x\) to some value
– Pick each variable in turn or randomly and resample \(P(x_i|x \neq i)\)

At equilibrium can assume \(x \sim P(x)\)

Consistent with \(x_{j \neq i} \sim P(x \neq i), \quad x_i \sim P(x_i | x_{j \neq i})\)

Proof of validity: a) check detailed balance for component update.
b) Metropolis–Hastings ‘proposals’ \(P(x_i|x_{j \neq i}) \Rightarrow\) accept with prob. 1
Apply a series of these operators. Don’t need to check acceptance.
Gibbs Sampling Implementation

Gibbs sampling benefits from few free choices and convenient features of conditional distributions:

- Conditionals with a few discrete settings can be explicitly normalized:

  \[
  P(x_i | x_j \neq i) \propto P(x_i, x_j \neq i) \\
  = \frac{P(x_i, x_j \neq i)}{\sum_{x_i'} P(x_i', x_j \neq i)} \leftarrow \text{this sum is small and easy}
  \]

- Continuous conditionals only univariate
  \Rightarrow \text{amenable to standard sampling methods.}

  - Inverse CDF sampling
  - Rejection sampling
  - Slice sampling
  - ...
Undirected Graphical Models

- This global Markov property implies a local Markov property:
  \[ p(x_A, x_C \mid x_B) = p(x_A \mid x_B)p(x_C \mid x_B) \]

- Graph Separation
- Conditional Independence

- This property generalizes temporal Markov processes, for which the past and future are independent conditioned on the present. For example, the undirected graph of Fig. 2.5 (a) implies the following conditional independencies, among others:

  \[
  p(x_1, x_2, x_5 \mid x_3, x_4) = p(x_1, x_2 \mid x_3, x_4) p(x_5 \mid x_3) \\
  p(x_1, x_4, x_5 \mid x_2, x_3) = p(x_1 \mid x_2, x_3) p(x_4 \mid x_2, x_3) p(x_5 \mid x_3)
  \]

- Practical benefits of Gibbs sampling algorithm:
  - Model and algorithm have same modular structure
  - Conditionals can often be evaluated quickly, because they depend only on the neighboring nodes
  - Exponential families offer further efficiency improvements, by caching and recursively updating sufficient statistics
Gibbs Sampling as Message Passing

- Consider a pairwise undirected graphical model:

\[
p(x) = \frac{1}{Z} \prod_{(s,t) \in E} \psi_{st}(x_s, x_t) \prod_{s \in V} \psi_s(x_s)
\]

- \(q_i(x_i) \propto \psi_i(x_i) \prod_{j \in \Gamma(i)} m_{ji}(x_i)\)

- \(\hat{x}_i \sim q_i(x_i)\) Draw single sample from marginal

- \(m_{ij}(x_j) \propto \psi_{ij}(\hat{x}_i, x_j)\) Use sample to extract a “slice” of pairwise potential

- Valid for discrete and continuous variables, although sampling step may be harder for continuous models

- General factor graphs have similar form
MCMC Implementation & Application

• The samples aren’t independent. Should we thin, only keep every $K$th sample?

• Arbitrary initialization means starting iterations are bad. Should we discard a “burn-in” period?

• Maybe we should perform multiple runs?

• How do we know if we have run for long enough?
Estimating Moments from Samples

Approximately independent samples can be obtained by *thinning*. However, all the samples can be used.

Use the simple Monte Carlo estimator on MCMC samples. It is:

— consistent
— unbiased if the chain has “burned in”

\[
\mathbb{E}_P[f] \approx \frac{1}{S} \sum_{s=1}^{S} f(x^{(s)})
\]

The correct motivation to thin: if computing \( f(x^{(s)}) \) is expensive
Probabilistic Mixture Models

\[ p(x_i \mid z_i, \mu, \Sigma) = \mathcal{N}(x_i \mid \mu_{z_i}, \Sigma_{z_i}) \]
Mixture Sampler Pseudocode

Given mixture weights $\pi^{(t-1)}$ and cluster parameters $\{\theta_k^{(t-1)}\}_k$ from the previous iteration, sample a new set of mixture parameters as follows:

1. Independently assign each of the $N$ data points $x_i$ to one of the $K$ clusters by sampling the indicator variables $z = \{z_i\}_{i=1}^N$ from the following multinomial distributions:

$$ z_i^{(t)} \sim \frac{1}{Z_i} \sum_{k=1}^K \pi^{(t-1)}_k f(x_i \mid \theta^{(t-1)}_k) \delta(z_i, k) \quad Z_i = \sum_{k=1}^K \pi^{(t-1)}_k f(x_i \mid \theta^{(t-1)}_k) $$

2. Sample new mixture weights according to the following Dirichlet distribution:

$$ \pi^{(t)} \sim \text{Dir}(N_1 + \alpha/K, \ldots, N_K + \alpha/K) \quad N_k = \sum_{i=1}^N \delta(z_i^{(t)}, k) $$

3. For each of the $K$ clusters, independently sample new parameters from the conditional distribution implied by those observations currently assigned to that cluster:

$$ \theta_k^{(t)} \sim p(\theta_k \mid \{x_i \mid z_i^{(t)} = k\}, \lambda) $$

When $\lambda$ defines a conjugate prior, this posterior distribution is given by Prop. 2.1.4.

**Proposition 2.1.4.** Let $p(x \mid \theta)$ denote an exponential family with canonical parameters $\theta$, and $p(\theta \mid \lambda)$ a family of conjugate priors defined as in eq. (2.28). Given $L$ independent samples $\{x^{(\ell)}\}_{\ell=1}^L$, the posterior distribution remains in the same family:

For each mixture component, posterior given assigned data

$$ p(\theta \mid x^{(1)}, \ldots, x^{(L)}, \lambda) = p(\theta \mid \bar{\lambda}) $$

(2.31)

$$ \bar{\lambda}_0 = \lambda_0 + L \quad \bar{\lambda}_a = \frac{\lambda_0 \lambda_a + \sum_{\ell=1}^L \phi_a(x^{(\ell)})}{\lambda_0 + L} \quad a \in \mathcal{A} $$

(2.32)
Mixture Sampler Pseudocode

Given mixture weights $\pi^{(t-1)}$ and cluster parameters $\{\theta_k^{(t-1)}\}_{k=1}^K$ from the previous iteration, sample a new set of mixture parameters as follows:

1. Independently assign each of the $N$ data points $x_i$ to one of the $K$ clusters by sampling the indicator variables $z = \{z_i\}_{i=1}^N$ from the following multinomial distributions:

   $$z_i^{(t)} \sim \frac{1}{Z_i} \sum_{k=1}^K \pi_k^{(t-1)} f(x_i \mid \theta_k^{(t-1)}) \delta(z_i, k)$$

   $$Z_i = \sum_{k=1}^K \pi_k^{(t-1)} f(x_i \mid \theta_k^{(t-1)})$$

2. Sample new mixture weights according to the following Dirichlet distribution:

   $$\pi^{(t)} \sim \text{Dir}(N_1 + \alpha/K, \ldots, N_K + \alpha/K)$$

   $$N_k = \sum_{i=1}^N \delta(z_i^{(t)}, k)$$

3. For each of the $K$ clusters, independently sample new parameters from the conditional distribution implied by those observations currently assigned to that cluster:

   $$\theta_k^{(t)} \sim p(\theta_k \mid \{x_i \mid z_i^{(t)} = k\}, \lambda)$$

   When $\lambda$ defines a conjugate prior, this posterior distribution is given by Prop. 2.1.4.

Compared to the EM algorithm for finite mixture models:

- Form same assignment indicator distributions as in E-step, but then draw a single sample from each distribution
- Sample, rather than taking mode, of parameter distributions
Snapshots of Mixture Gibbs Sampler

Initialization A

Initialization B

2 Iterations

10 Iterations

50 Iterations

\[ \log p(x | \pi, \theta) = -539.17 \]

\[ \log p(x | \pi, \theta) = -404.18 \]

\[ \log p(x | \pi, \theta) = -397.40 \]

\[ \log p(x | \pi, \theta) = -497.77 \]

\[ \log p(x | \pi, \theta) = -454.15 \]

\[ \log p(x | \pi, \theta) = -442.89 \]
MCMC: Mixing Diagnostics

**Autocovariance:** Empirical covariance of values produced by MCMC method, versus iteration lag (spacing)

- Small autocovariances are necessary, but not sufficient, to demonstrate mixing to the target distribution
- Fairly reliable for unimodal posteriors, but *very misleading more generally*

**Trace Plot:** Value of some “interesting” summary statistic, versus MCMC iteration
MCMC & Computational Resources

Best practical option: A few (> 1) initializations for as many iterations as possible
Rao-Blackwellized Estimation

- Basic Monte Carlo estimation for joint distribution of $x$, $z$:

$$ (x^{(l)}, z^{(l)}) \sim p(x, z) \quad \ell = 1, 2, \ldots, L $$

$$ E_p[f(x, z)] = \int_x \int_z f(x, z)p(x, z) \, dx \, dz \approx \frac{1}{L} \sum_{\ell=1}^L f(x^{(\ell)}, z^{(\ell)}) = E_{\tilde{p}}[f(x, z)] $$

- But suppose that the conditional distribution $p(x \mid z)$ is tractable:

$$ E_p[f(x, z)] = \int_z \int_x f(x, z)p(x \mid z) \, p(z) \, dx \, dz $$

$$ = \int_z \left[ \int_x f(x, z)p(x \mid z) \, dx \right] p(z) \, dz $$

$$ \approx \frac{1}{L} \sum_{\ell=1}^L \int_x f(x, z^{(\ell)})p(x \mid z^{(\ell)}) \, dx = E_{\tilde{p}}[E_p[f(x, z) \mid z]] $$

- Should we expect this estimator to be more accurate?
Conditional vs Unconditional Variance

- The Rao-Blackwell Theorem, which was classically used to reduce the variance of estimators, is based on this identity:

\[ \text{Theorem 2.4.1 (Rao-Blackwell). Let } x \text{ and } z \text{ be dependent random variables, and } f(x, z) \text{ a scalar statistic. Consider the marginalized statistic } \mathbb{E}_x[f(x, z) \mid z], \text{ which is a function solely of } z. \text{ The unconditional variance } \text{Var}_{xz}[f(x, z)] \text{ is then related to the variance of the marginalized statistic as follows:} \]

\[
\text{Var}_{xz}[f(x, z)] = \text{Var}_z[\mathbb{E}_x[f(x, z) \mid z]] + \mathbb{E}_z[\text{Var}_x[f(x, z) \mid z]] \\
\geq \text{Var}_z[\mathbb{E}_x[f(x, z) \mid z]] 
\]

*Basic estimator* \hspace{1cm} *RB estimator* \hspace{1cm} *non-negative*

- Applications in Monte Carlo methods:
  - Given output of any “standard” MCMC method, process to produce more efficient estimators
  - Analytically marginalize, or collapse, some variables from the model and derive Gibbs sampler for this collapsed representation
Collapsed Sampling Algorithms

\[ z_i \sim \text{Cat}(\pi) \]
\[ x_i \sim F(\theta_{z_i}) \]
\[ \theta_k \sim G(\beta) \]

Conjugate priors allow exact marginalization of parameters, to make an equivalent model with fewer variables.
Given previous cluster assignments $z^{(t-1)}$, sequentially sample new assignments as follows:

1. Sample a random permutation $\tau(\cdot)$ of the integers $\{1, \ldots, N\}$.
2. Set $z = z^{(t-1)}$. For each $i \in \{\tau(1), \ldots, \tau(N)\}$, sequentially resample $z_i$ as follows:
   
   (a) For each of the $K$ clusters, determine the predictive likelihood
   
   $$f_k(x_i) = p(x_i \mid \{x_j \mid z_j = k, j \neq i\}, \lambda)$$

   This likelihood can be computed from cached sufficient statistics via Prop. 2.1.4.
   
   (b) Sample a new cluster assignment $z_i$ from the following multinomial distribution:

   $$z_i \sim \frac{1}{Z_i} \sum_{k=1}^{K} (N_{k}^{-i} + \alpha/K) f_k(x_i) \delta(z_i, k) \quad Z_i = \sum_{k=1}^{K} (N_{k}^{-i} + \alpha/K) f_k(x_i)$$

   $N_{k}^{-i}$ is the number of other observations assigned to cluster $k$ (see eq. (2.162)).
   
   (c) Update cached sufficient statistics to reflect the assignment of $x_i$ to cluster $z_i$.

3. Set $z^{(t)} = z$. Optionally, mixture parameters may be sampled via steps 2–3 of Alg. 2.1.

\begin{align}
   p(\theta \mid x^{(1)}, \ldots, x^{(L)}, \lambda) &= p(\theta \mid \bar{\lambda}) \\
   \bar{\lambda}_0 &= \lambda_0 + L \quad \bar{\lambda}_a &= \frac{\lambda_0 \lambda_a + \sum_{\ell=1}^{L} \phi_a(x^{(\ell)})}{\lambda_0 + L} \quad a \in A
\end{align}

Integrating over $\Theta$, the log–likelihood of the observations can then be compactly written using the normalization constant of eq. (2.29):

$$\log p(x^{(1)}, \ldots, x^{(L)} \mid \lambda) = \Omega(\bar{\lambda}) - \Omega(\lambda) + \sum_{\ell=1}^{L} \log \nu(x^{(\ell)})$$
Gibbs: Representation and Mixing

Multiple Initializations

Quantiles of 100 Chains

Standard Gibbs:  Alternatively sample assignments, parameters
Collapsed Gibbs:  Marginalize parameters, sample assignments