Approx. Inference via Sampling (Contd): Metropolis Hastings and Gibbs Sampling

CS772A: Probabilistic Machine Learning

Piyush Rai

Plan for today

- MCMC algorithms
  - Metropolis Hastings (MH)
  - Gibbs sampling (special case of MH)
  - Some examples of Gibbs sampling

The MH Sampling Algorithm

- Initialize $\boldsymbol{z}^{(1)}$ randomly
- For $\ell = 1,2, \dots, L$
  - Sample $\boldsymbol{z}^* \sim q(\boldsymbol{z}^*|\boldsymbol{z}^{(\ell)})$ and $u \sim \text{Unif}(0,1)$
  - Compute acceptance probability

$$A(\boldsymbol{z}^*, \boldsymbol{z}^{(\ell)}) = \min\left(1, \frac{\tilde{p}(\boldsymbol{z}^*)q(\boldsymbol{z}^{(\ell)}|\boldsymbol{z}^*)}{\tilde{p}(\boldsymbol{z}^{(\ell)})q(\boldsymbol{z}^*|\boldsymbol{z}^{(\ell)})}\right)$$

  - If $A\left(\boldsymbol{z}^*, \boldsymbol{z}^{(\ell)}\right) > u$

$$\boldsymbol{z}^{(\ell+1)} = \boldsymbol{z}^*$$

Meaning accepting $\boldsymbol{z}^*$ with probability $A\left(\boldsymbol{z}^*, \boldsymbol{z}^{(\ell)}\right)$

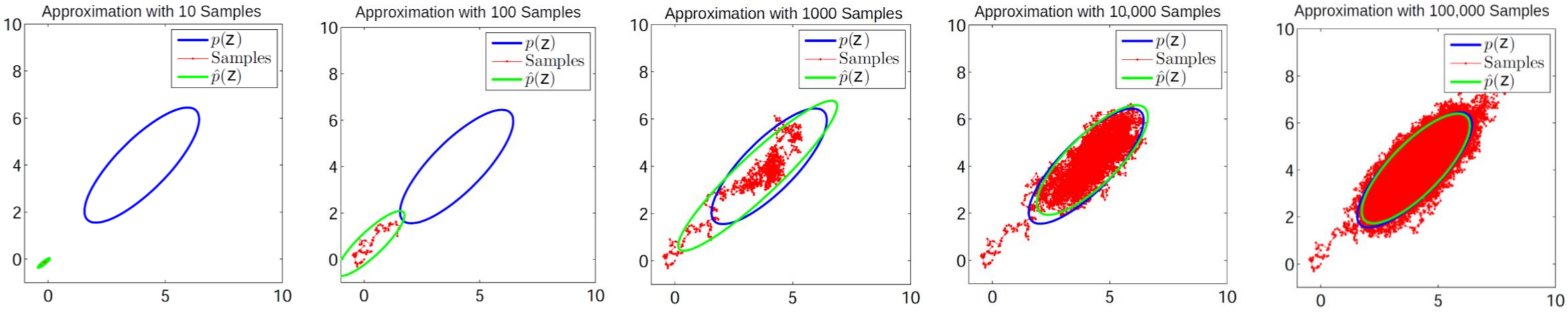
  - Else

$$\boldsymbol{z}^{(\ell+1)} = \boldsymbol{z}^{(\ell)}$$

MH Sampling in Action: A Toy Example..

- Target distribution $p(\mathbf{z}) = \mathcal{N}\left(\begin{bmatrix}4\\4\end{bmatrix}, \begin{bmatrix}1 & 0.8\\0.8 & 1\end{bmatrix}\right)$

- Proposal distribution $q(\mathbf{z}^{(t)}|\mathbf{z}^{(t-1)}) = \mathcal{N}\left(\mathbf{z}^{(t-1)}, \begin{bmatrix}0.01 & 0\\0 & 0.01\end{bmatrix}\right)$

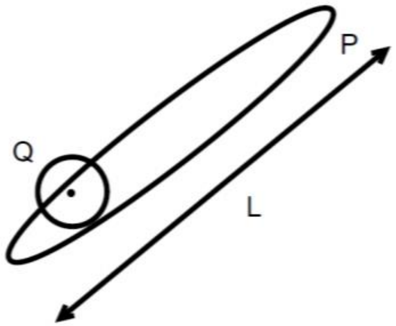


MH Sampling: Some Comments

- If prop. distrib. is symmetric, we get Metropolis Sampling algo (Metropolis, 1953) with

$$A(\mathbf{z}^*, \mathbf{z}^{(\tau)}) = \min\left(1, \frac{\tilde{p}(\mathbf{z}^*)}{\tilde{p}(\mathbf{z}^{(\tau)})}\right)$$

- Some limitations of MH sampling
  - Can sometimes have very slow convergence (also known as slow "mixing")



$Q(\mathbf{z}|\mathbf{z}^{(\tau)}) = \mathcal{N}(\mathbf{z}|\mathbf{z}^{(\tau)}, \sigma^2\mathbf{I})$

σ large $\Rightarrow$ many rejections

σ small $\Rightarrow$ slow diffusion

$\sim \left(\frac{L}{\sigma}\right)^2$ iterations required for convergence

  - Computing acceptance probability can be expensive*, e.g., if $p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{Z_p}$ is some target posterior then $\tilde{p}(\mathbf{z})$ would require computing likelihood on all the data points (expensive)

*Austerity in MCMC Land: Cutting the Metropolis-Hastings Budget (Korattikara et al, 2014), Firefly Monte Carlo: Exact MCMC with Subsets of Data {(Maclaurin and Adams, 2015)

Gibbs Sampling (Geman & Geman, 1984)

- Goal: Sample from a joint distribution $p(\boldsymbol{z})$ where $\boldsymbol{z} = [z_1, z_2, \dots, z_M]$
- Suppose we can't sample from $p(\boldsymbol{z})$ but can sample from each conditional $p(z_i|\boldsymbol{z}_{-i})$
  - In Bayesian models, can be done easily if we have a locally conjugate model
- For Gibbs sampling, the proposal is the conditional distribution $p(z_i|\boldsymbol{z}_{-i})$
- Gibbs sampling samples from these conditionals in a cyclic order

Hence no need to compute it

- Gibbs sampling is equivalent to MH sampling with acceptance prob. = 1

$$A(\boldsymbol{z}^*, \boldsymbol{z}) = \frac{p(\boldsymbol{z}^*)q(\boldsymbol{z}|\boldsymbol{z}^*)}{p(\boldsymbol{z})q(\boldsymbol{z}^*|\boldsymbol{z})} = \frac{p(z_i^*|\boldsymbol{z}^*_{-i})p(\boldsymbol{z}^*_{-i})p(z_i|\boldsymbol{z}^*_{-i})}{p(z_i|\boldsymbol{z}_{-i})p(\boldsymbol{z}_{-i})p(z_i^*|\boldsymbol{z}_{-i})} = 1$$

where we use the fact that $\boldsymbol{z}^*_{-i} = \boldsymbol{z}_{-i}$

Since only one component is changed at a time

Gibbs Sampling: Sketch of the Algorithm

- $\boldsymbol{M}$: Total number of variables, $\boldsymbol{T}$: number of Gibbs sampling iterations

1. Initialize $\{z_i : i = 1, \ldots, M\}$
2. For $\tau = 1, \ldots, T$:
   - Sample $z_1^{(\tau+1)} \sim p(z_1 | z_2^{(\tau)}, z_3^{(\tau)}, \ldots, z_M^{(\tau)})$.
   - Sample $z_2^{(\tau+1)} \sim p(z_2 | z_1^{(\tau+1)}, z_3^{(\tau)}, \ldots, z_M^{(\tau)})$.

     $\vdots$
   - Sample $z_j^{(\tau+1)} \sim p(z_j | z_1^{(\tau+1)}, \ldots, z_{j-1}^{(\tau+1)}, z_{j+1}^{(\tau)}, \ldots, z_M^{(\tau)})$.

     $\vdots$
   - Sample $z_M^{(\tau+1)} \sim p(z_M | z_1^{(\tau+1)}, z_2^{(\tau+1)}, \ldots, z_{M-1}^{(\tau+1)})$.

Assuming $\boldsymbol{z} = [z_1, z_2, \ldots, z_M]$

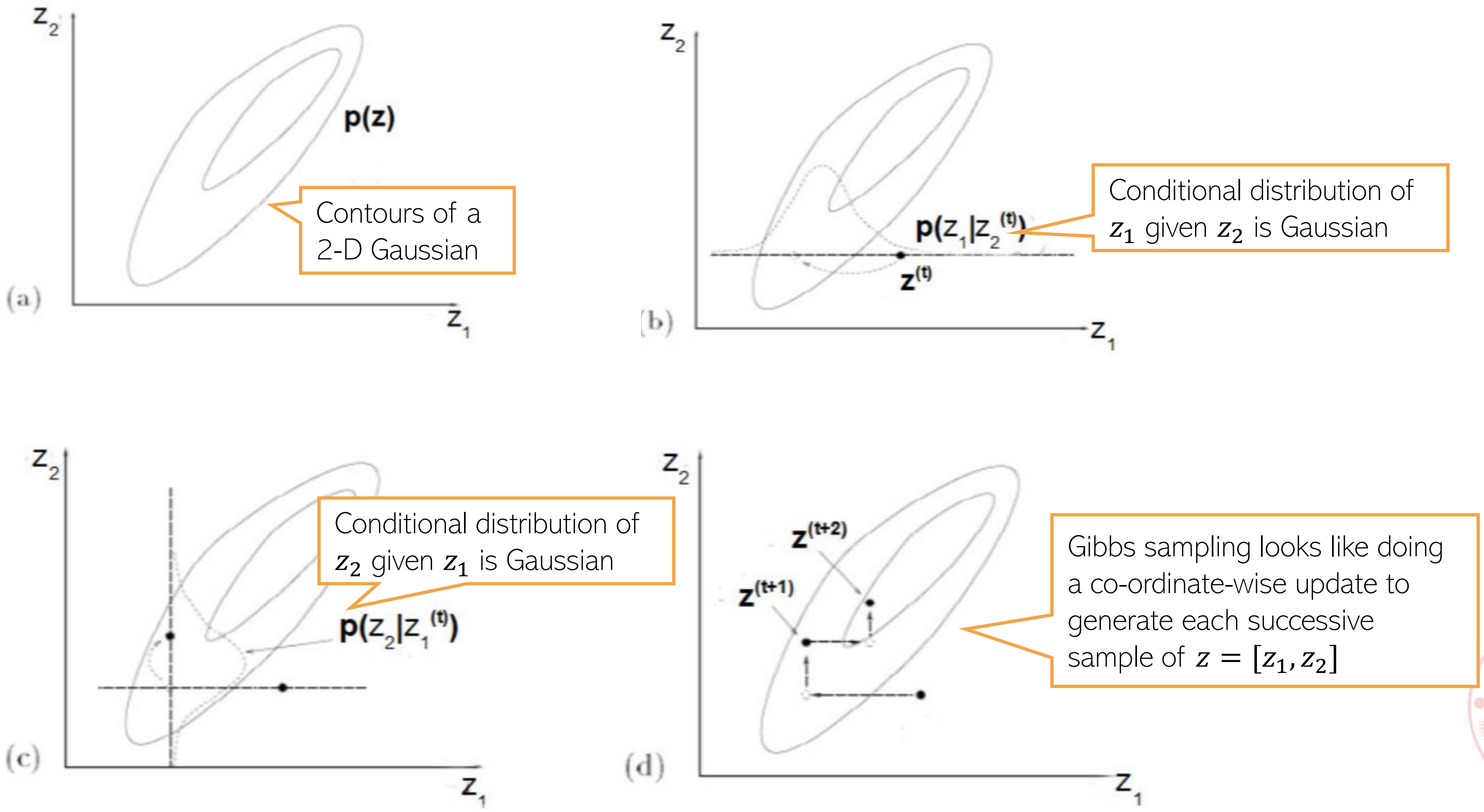
CP of each component of $\boldsymbol{z}$ uses the most recent values (from this or the previous iteration) of all the other components

Each iteration will give us one sample $\boldsymbol{z}^{(\tau)}$ of $\boldsymbol{z} = [z_1, z_2, \ldots, z_M]$

- Note: Order of updating the variables usually doesn't matter (but see "Scan Order in Gibbs Sampling: Models in Which it Matters and Bounds on How Much" from NIPS 2016)

Gibbs Sampling: A Simple Example

- Can sample from a 2-D Gaussian using 1-D Gaussians



Gibbs Sampling: Some Comments

- One of the most popular MCMC algorithms
- Very easy to derive and implement for locally conjugate models
- Many variations exist, e.g.,
  - Blocked Gibbs: sample more than one component jointly (sometimes possible)
  - Rao-Blackwellized Gibbs: Can collapse (i.e., integrate out) the unneeded components while sampling. Also called "collapsed" Gibbs sampling
  - MH within Gibbs: If CPs are not easy to sample distributions
- Instead of sampling from CPs, an alternative is to use the mode of the CPs
  - Called the "Iterative Conditional Mode" (ICM) algorithm
  - ICM doesn't give the posterior though – it's more like ALT-OPT to get (approx) MAP estimate

Recap: Gibbs Sampling

- An instance of MH sampling where the acceptance probability = 1
- Based on sampling $\boldsymbol{z}$ one "component" at a time with proposal = conditional distr.

Gibbs Sampling

Initialize $\mathbf{z}^{(0)} = [z_1^{(0)}, z_2^{(0)}, \ldots, z_M^{(0)}]$ randomly

For $\ell = 1, \ldots, L$

- Sample $\mathbf{z}^{(\ell)}$ by sampling one component at a time (usually cyclic manner)

$$
\begin{aligned}
z_1^{(\ell)} &\sim p(z_1 | z_2^{(\ell-1)}, z_3^{(\ell-1)}, \ldots, z_M^{(\ell-1)}) \\
z_2^{(\ell)} &\sim p(z_2 | z_1^{(\ell)}, z_3^{(\ell-1)}, \ldots, z_M^{(\ell-1)}) \\
&\vdots \\
z_{M-1}^{(\ell)} &\sim p(z_{M-1} | z_1^{(\ell)}, \ldots, z_{M-2}^{(\ell)}, z_M^{(\ell-1)}) \\
z_M^{(\ell)} &\sim p(z_M | z_1^{(\ell)}, z_2^{(\ell)}, \ldots, z_{M-1}^{(\ell)})
\end{aligned}
$$

$\boldsymbol{z}^{(\ell)}$

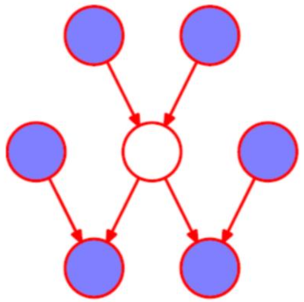
In practice, we won't use all the L samples to approximate the target distribution $p(\boldsymbol{z})$ since there will be a burn-in phase and thinning as well

Denoting the collected samples by $\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}, \ldots, \boldsymbol{z}^{(S)}$, the posterior approximation will be the empirical distribution defined by these samples

- Very easy to derive if the conditional distributions are easy to obtain

Deriving A Gibbs Sampler: The General Recipe

- Suppose the target is an intractable posterior $p(\boldsymbol{Z}|\boldsymbol{X})$ where $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_M]$
- Gibbs sampling requires the conditional posteriors $p(\boldsymbol{z}_m|\boldsymbol{Z}_{-m}, \boldsymbol{X})$
- In general, $p(\boldsymbol{z}_m|\boldsymbol{Z}_{-m}, \boldsymbol{X}) \propto p(\boldsymbol{z}_m)p(\boldsymbol{X}|\boldsymbol{z}_m, \boldsymbol{Z}_{-m})$ where $\boldsymbol{Z}_{-m}$ is assumed "known"
- If $p(\boldsymbol{z}_m)$ and $p(\boldsymbol{X}|\boldsymbol{z}_m, \boldsymbol{Z}_{-m})$ are conjugate, the above CP is straightforward to obtain
- Another way to get each CP $p(\boldsymbol{z}_m|\boldsymbol{Z}_{-m}\boldsymbol{X})$ is by following this
  - Write down the expression of $p(\boldsymbol{X}, \boldsymbol{Z})$
  - Only terms that contain $\boldsymbol{z}_m$ needed to get CP of $\boldsymbol{z}_m$ (up to a prop const)



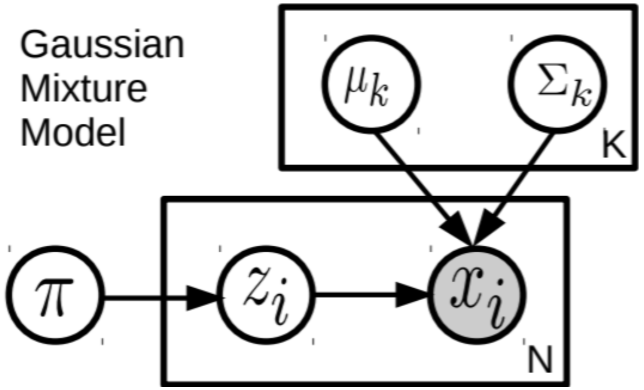
Markov Blanket

- In $p(\boldsymbol{z}_m|\boldsymbol{Z}_{-m}, \boldsymbol{X})$, we only need to condition on terms in Markov Blanket of $\boldsymbol{z}_m$
  - Markov Blanket of a variable: Its parents, children, and other parents of its children
  - Very useful in deriving CP

Gibbs Sampling: An Example

- The CPs for the Gibbs sampler for a GMM are as shown in green rectangles below

Gaussian Mixture Model



Can verify that Markov Blanket property holds for each CP

Joint distribution of data and unknowns

$$
\begin{aligned}
p(\mathbf{x}, \mathbf{z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) &= p(\mathbf{x}|\mathbf{z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) p(\mathbf{z}|\boldsymbol{\pi}) p(\boldsymbol{\pi}) \prod_{k=1}^{K} p(\boldsymbol{\mu}_k) p(\boldsymbol{\Sigma}_k) \\
&= \left(\prod_{i=1}^{N} \prod_{k=1}^{K} (\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k))^{\mathbb{I}(z_i=k)} \right) \times \\
& \quad \mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha}) \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{\mu}_k|\mathbf{m}_0, \mathbf{V}_0) \mathrm{IW}(\boldsymbol{\Sigma}_k|\mathbf{S}_0, \nu_0)
\end{aligned}
$$

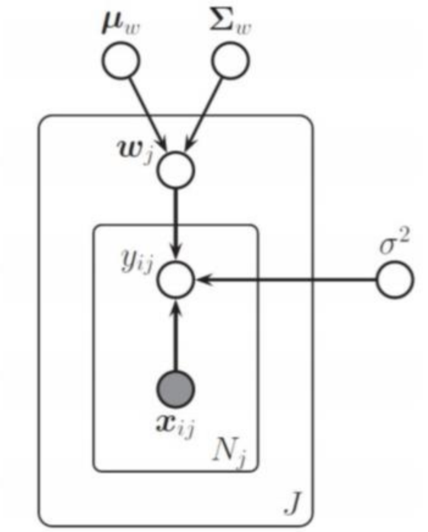
$$
p(z_i = k|\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) \propto \pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)
$$

$$
p(\boldsymbol{\pi}|\mathbf{z}) = \mathrm{Dir}(\{\alpha_k + \sum_{i=1}^{N} \mathbb{I}(z_i = k)\}_{k=1}^{K})
$$

$$
\begin{aligned}
p(\boldsymbol{\mu}_k|\boldsymbol{\Sigma}_k, \mathbf{z}, \mathbf{x}) &= \mathcal{N}(\boldsymbol{\mu}_k|\mathbf{m}_k, \mathbf{V}_k) \\
\mathbf{V}_k^{-1} &= \mathbf{V}_0^{-1} + N_k \boldsymbol{\Sigma}_k^{-1} \\
\mathbf{m}_k &= \mathbf{V}_k(\boldsymbol{\Sigma}_k^{-1} N_k \overline{\mathbf{x}}_k + \mathbf{V}_0^{-1} \mathbf{m}_0) \\
N_k &\triangleq \sum_{i=1}^{N} \mathbb{I}(z_i = k) \\
\overline{\mathbf{x}}_k &\triangleq \frac{\sum_{i=1}^{N} \mathbb{I}(z_i = k) \mathbf{x}_i}{N_k}
\end{aligned}
$$

$$
\begin{aligned}
p(\boldsymbol{\Sigma}_k|\boldsymbol{\mu}_k, \mathbf{z}, \mathbf{x}) &= \mathrm{IW}(\boldsymbol{\Sigma}_k|\mathbf{S}_k, \nu_k) \\
\mathbf{S}_k &= \mathbf{S}_0 + \sum_{i=1}^{N} \mathbb{I}(z_i = k)(\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \\
\nu_k &= \nu_0 + N_k
\end{aligned}
$$

Gibbs Sampling: Another Example



J schools Regression Problem

$$p\left(\boldsymbol{Y}, \{\boldsymbol{w}_j\}_{j=1}^J, \boldsymbol{\mu}_w, \boldsymbol{\Sigma}_w, \sigma^2 \middle| \boldsymbol{X}\right)$$

Joint distribution of data and unknowns

$$= \left(\prod_{j=1}^J \prod_{i=1}^{N_j} p(y_{ij}|\boldsymbol{x}_{ij}, \boldsymbol{w}_j, \sigma^2) p(\boldsymbol{w}_j|\boldsymbol{\mu}_w, \boldsymbol{\Sigma}_w)\right) p(\boldsymbol{\mu}_w) p(\boldsymbol{\Sigma}_w) p(\sigma^2)$$

$$= \left(\prod_{j=1}^J \prod_{i=1}^{N_j} \mathcal{N}(y_{ij}|\boldsymbol{w}_j^\top \boldsymbol{x}_{ij}, \sigma^2) \mathcal{N}(\boldsymbol{w}_j|\boldsymbol{\mu}_w, \boldsymbol{\Sigma}_w)\right)$$

$$\mathcal{N}(\boldsymbol{\mu}_w|\boldsymbol{\mu}_0, \mathbf{V}_0) \mathrm{IW}(\boldsymbol{\Sigma}_w|\boldsymbol{\eta}_0, \mathbf{S}_0^{-1})\ \mathrm{IG}(\sigma^2|\nu_0/2, \nu_0\sigma_0^2/2)$$

Can verify that Markov Blanket property holds for each CP

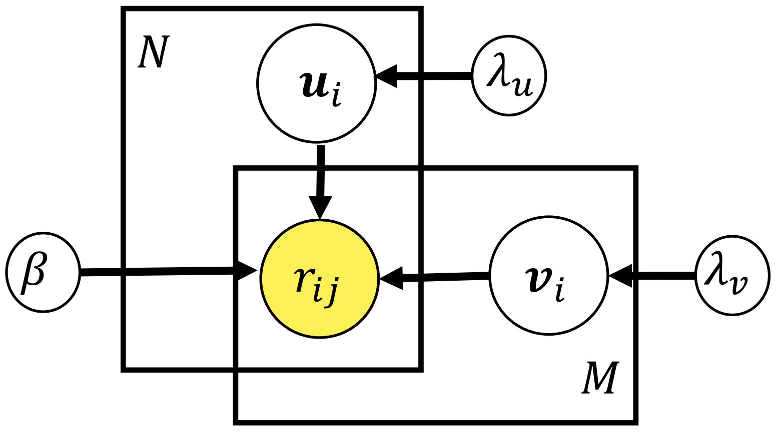
$$\begin{aligned} p(\mathbf{w}_j|\mathcal{D}_j, \boldsymbol{\theta}) &= \mathcal{N}(\mathbf{w}_j|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \\ \boldsymbol{\Sigma}_j^{-1} &= \boldsymbol{\Sigma}^{-1} + \mathbf{X}_j^T \mathbf{X}_j/\sigma^2 \\ \boldsymbol{\mu}_j &= \boldsymbol{\Sigma}_j(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \mathbf{X}_j^T \mathbf{y}_j/\sigma^2) \end{aligned}$$

$$\begin{aligned} p(\boldsymbol{\mu}_w|\mathbf{w}_{1:J}, \boldsymbol{\Sigma}_w) &= \mathcal{N}(\boldsymbol{\mu}|\boldsymbol{\mu}_N, \boldsymbol{\Sigma}_N) \\ \boldsymbol{\Sigma}_N^{-1} &= \mathbf{V}_0^{-1} + J\boldsymbol{\Sigma}^{-1} \\ \boldsymbol{\mu}_N &= \boldsymbol{\Sigma}_N(\mathbf{V}_0^{-1}\boldsymbol{\mu}_0 + J\boldsymbol{\Sigma}^{-1}\overline{\mathbf{w}}) \\ \overline{\mathbf{w}} &= \frac{1}{J}\sum_j \mathbf{w}_j \end{aligned}$$

$$\begin{aligned} p(\boldsymbol{\Sigma}_w|\boldsymbol{\mu}_w, \mathbf{w}_{1:J}) &= \mathrm{IW}((\mathbf{S}_0 + \mathbf{S}_\mu)^{-1}, \eta_0 + J) \\ \mathbf{S}_\mu &= \sum_j (\mathbf{w}_j - \boldsymbol{\mu}_w)(\mathbf{w}_j - \boldsymbol{\mu}_w)^T \end{aligned}$$

$$\begin{aligned} p(\sigma^2|\mathcal{D}, \mathbf{w}_{1:J}) &= \mathrm{IG}([\nu_0 + N]/2, [\nu_0\sigma_0^2 + \mathrm{SSR}(\mathbf{w}_{1:J})]/2) \\ \mathrm{SSR}(\mathbf{w}_{1:J}) &= \sum_{j=1}^J \sum_{i=1}^{N_j} (y_{ij} - \mathbf{w}_j^T \mathbf{x}_{ij})^2 \end{aligned}$$

Gibbs Sampling: One More Example



Bayesian Matrix Factorization

$$p(\mathbf{R}, \{\boldsymbol{u}_i\}_{i=1}^N, \{\boldsymbol{v}_j\}_{j=1}^M, \lambda_u, \lambda_v, \beta)$$

Joint distribution of data and unknowns

Assuming even the hyperparams to be unknown

$$= \prod_{(i,j)\in\Omega} p(r_{ij}|\boldsymbol{u}_i, \boldsymbol{v}_j, \beta) \prod_i p(\boldsymbol{u}_i|\lambda_u) \prod_j p(\boldsymbol{v}_j|\lambda_v)\, p(\lambda_u)p(\lambda_v)p(\beta)$$

Can also use non-zero mean and full cov matrix for $\boldsymbol{u}_i, \boldsymbol{v}_j$, with Gaussian and Wishart priors respectively*

$$= \prod_{(i,j)\in\Omega} \mathcal{N}(r_{ij}|\boldsymbol{u}_i^\top \boldsymbol{v}_j, \beta) \prod_i \mathcal{N}(\boldsymbol{u}_i|0, \lambda_u^{-1}\mathbf{I}) \prod_j \mathcal{N}(\boldsymbol{v}_j|0, \lambda_v^{-1}\mathbf{I})$$

$$\text{Gamma}(\lambda_u|a,b)\text{Gamma}(\lambda_v|c,d)\text{Gamma}(\beta|e,f)$$

Can verify that Markov Blanket property holds for each CP

$$p(\mathbf{u}_i|\mathbf{R},\mathbf{V}) = \mathcal{N}(\mathbf{u}_i|\boldsymbol{\mu}_{u_i}, \boldsymbol{\Sigma}_{u_i})$$
$$\boldsymbol{\Sigma}_{u_i} = (\lambda_u \mathbf{I} + \beta \textstyle\sum_{j:(i,j)\in\Omega} \mathbf{v}_j \mathbf{v}_j^\top)^{-1}$$
$$\boldsymbol{\mu}_{u_i} = \boldsymbol{\Sigma}_{u_i}(\beta \textstyle\sum_{j:(i,j)\in\Omega} r_{ij}\mathbf{v}_j)$$

$$p(\mathbf{v}_j|\mathbf{R},\mathbf{U}) = \mathcal{N}(\mathbf{v}_j|\boldsymbol{\mu}_{v_j}, \boldsymbol{\Sigma}_{v_j})$$
$$\boldsymbol{\Sigma}_{v_j} = (\lambda_v \mathbf{I} + \beta \textstyle\sum_{i:(i,j)\in\Omega} \mathbf{u}_i \mathbf{u}_i^\top)^{-1}$$
$$\boldsymbol{\mu}_{v_j} = \boldsymbol{\Sigma}_{v_j}(\beta \textstyle\sum_{i:(i,j)\in\Omega} r_{ij}\mathbf{u}_i)$$

$$p(\lambda_u|\mathbf{U}) = \text{Gamma}(\lambda_u|a + 0.5 * NK, b + 0.5 * \sum_{i=1}^N \boldsymbol{u}_i^\top \boldsymbol{u}_i)$$
$$p(\lambda_v|\mathbf{V}) = \text{Gamma}(\lambda_v|c + 0.5 * MK, d + 0.5 * \sum_{j=1}^M \boldsymbol{v}_j^\top \boldsymbol{v}_j)$$

$$p(\beta|\mathbf{R},\mathbf{U},\mathbf{V}) = \text{Gamma}(\beta|e + 0.5 * |\Omega|, f + 0.5 * \textstyle\sum_{i,j\in\Omega}(r_{ij} - \boldsymbol{u}_i^\top \boldsymbol{v}_j)^2)$$

Ω denotes the indices that are observed in the ratings matrix

*Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo (Salakhutdinov and Mnih, 2008)

MCMC: Some Other Aspects

Using the Samples to make Predictions

- Using the S samples $\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)}, \ldots, \boldsymbol{Z}^{(S)}$, our approx. $p(\boldsymbol{Z}) \approx \frac{1}{S} \sum_{s=1}^{S} \delta_{\boldsymbol{Z}^{(s)}}(\boldsymbol{Z})$

- Any expectation that depends on $p(\boldsymbol{Z})$ be approximated as

$$\mathbb{E}[f(\boldsymbol{Z})] = \int f(\boldsymbol{Z}) p(\boldsymbol{Z}) d\boldsymbol{Z} \approx \frac{1}{S} \sum_{s=1}^{S} f(\boldsymbol{Z}^{(s)})$$

- For Bayesian lin. reg., assuming $\boldsymbol{w}, \beta, \lambda$ to be unknown, the PPD approx. will be

Joint posterior over all unknowns

Thus, in this case, the PPD is a sum of S Gaussians

$$\int p(y_* | \boldsymbol{x}_*, \boldsymbol{w}, \beta) p(\boldsymbol{w}, \beta, \lambda | \boldsymbol{X}, \boldsymbol{y}) d\boldsymbol{w} d\beta d\lambda \approx \frac{1}{S} \sum_{s=1}^{S} p\left(y_* \middle| \boldsymbol{x}_*, \boldsymbol{w}^{(s)}, \beta^{(s)}\right)$$

Sampling based approximation of PPD

Mean and variance of y_* can be computed using sum of Gaussian properties

Mean: $\mathbb{E}[y_*] = \frac{1}{S} \sum_{s=1}^{S} {\boldsymbol{w}^{(s)}}^\top \boldsymbol{x}_*$

Variance: Exercise! Use definition of variance and use Monte-Carlo approximation

- Sampling based approx. for PPD of other models can also be obtained likewise

Sampling Methods: Label Switching Issue

- Suppose we are given samples $\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)}, \dots, \boldsymbol{Z}^{(S)}$ from the posterior $p(\boldsymbol{Z}|\boldsymbol{X})$
- We can't always simply "average" them to get the "posterior mean" $\bar{\boldsymbol{Z}}$
- Why: Non-identifiability of latent vars in models with multiple equival. posterior modes
- Example: In clustering via GMM, the likelihood is invariant to how we label clusters
  - What we call cluster 1 in one sample may be cluster 2 in the next sample
  - Say, in GMM, $z_n^{(1)} = [1,0]$ and $z_n^{(2)} = [0,1]$, both samples imply the same
  - Averaging will give $\bar{z}_n = [0.5,0.5]$, which is incorrect

One sample may be from near one of the modes and the other may be from near the other mode

- Quantities not affected by permutations of dims of $\boldsymbol{Z}$ can be safely averaged
  - E.g., probability that two points belong to the same cluster (e.g., in GMM)
  - Predicting the mean of an entry r_{ij} in matrix factorization $\frac{1}{S}\sum_{s=1}^{S} \boldsymbol{u}_i^{(s)\top} \boldsymbol{v}_j^{(s)}$

Changes in order of entries in these $K \times 1$ vectors across different samples doesn't affect the inner product

MCMC: Some Practical Aspects

- Choice of proposal distribution is important
  - For MH sampling, Gaussian proposal is popular when $\boldsymbol{z}$ is continuous, e.g.,

$$q(\boldsymbol{z}|\boldsymbol{z}^{(\ell-1)}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{z}^{(\ell-1)}, \mathbf{H})$$

Hessian at the MAP of the target distribution

Change at each iter

  - Other options: Mixture of proposal distributions, data-driven or adaptive proposals
- Autocorrelation. Can show that when approximating $f^* = \mathbb{E}[f]$ using $\{\boldsymbol{Z}^{(s)}\}_{s=1}^{S}$

$\bar{f} = \frac{1}{S}\sum_{s=1}^{S} f_s$

Monte Carlo assumes uncorrelated samples

Value of f using s^{th} MCMC sample

Basically measures what fractions of the total samples are uncorrelated. Want it to be close to 1

$$\text{var}_{MCMC}[\bar{f}] = \text{var}_{MC}[\bar{f}] + \frac{1}{S^2}\sum_{s\neq t}\mathbb{E}[(f_s - f^*)(f_t - f^*)]$$

$$\text{Effective Sample Size (ESS)} = \frac{\text{var}_{MC}[f]}{\text{var}_{MCMC}[f]}$$

- Autocorrelation function (ACF) at lag t: $\rho_t = \frac{\frac{1}{S-t}\sum_{s=1}^{S-t}(f_s-\bar{f})(f_{s+t}-\bar{f})}{\frac{1}{S-1}\sum_{s=1}^{S}(f_s-\bar{f})^2}$

Lower is better

- Multiple Chains: Run multiple chains, take union of generated samples

Approximate Inference: VI vs Sampling

- VI approximates a posterior distribution $\boldsymbol{p(Z|X)}$ by another distribution $\boldsymbol{q(Z|\phi)}$
- Sampling uses $\boldsymbol{S}$ samples $\boldsymbol{Z^{(1)}, Z^{(2)}, \dots, Z^{(S)}}$ to approximate $\boldsymbol{p(Z|X)}$
- Sampling can be used within VI (ELBO approx using Monte-Carlo)
- In terms of "comparison" between VI and sampling, a few things to be noted
  - Convergence: VI only has local convergence, sampling (in theory) can give exact posterior
  - Storage: Sampling based approx needs to storage all samples, VI only needs var. params $\boldsymbol{\phi}$
  - Prediction Cost: Sampling always requires Monte-Carlo avging for posterior predictive; with VI, sometimes we can get closed form posterior predictive

PPD if using sampling: $p(x_*|X) = \int p(x_*|Z)p(Z|X)dZ \approx \frac{1}{S}\sum_{s=1}^{S} p\left(x_*|Z^{(s)}\right)$

PPD if using VI: $p(x_*|X) = \int p(x_*|Z)p(Z|X)dZ \approx \int p(x_*|Z)q(Z|\phi)dZ$

Compressing the S samples into something more compact

- There is some work on "compressing" sampling-based approximations*

*"Compact approximations to Bayesian predictive distributions" by Snelson and Ghaharamani, 2005; and "Bayesian Dark Knowledge" by Korattikara et al, 2015

Coming Up Next

- Avoiding the random-walk behavior of MCMC
  - Using gradient information of the posterior
- Scalable MCMC methods