Markov Chain Monte Carlo (MCMC) Sampling

CS772A: Probabilistic Machine Learning

Piyush Rai

Markov Chain Monte Carlo (MCMC)

If the target is a posterior, it will be conditioned on data, i.e., $p(\boldsymbol{z}|\boldsymbol{x})$

- Goal: Generate samples from some target distribution $p(\boldsymbol{z}) = \frac{\tilde{p}(\boldsymbol{z})}{Z_p}$

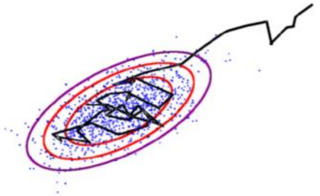
$\boldsymbol{z}$ usually is high-dim

- Assume we can evaluate $p(\boldsymbol{z})$ at least up to a proportionality constant

Means we can at least evaluate $\tilde{p}(\boldsymbol{z})$

- MCMC uses a Markov Chain which, when converged, starts giving samples from $p(z)$

$$\underbrace{\mathbf{z}^{(1)} \rightarrow \mathbf{z}^{(2)} \rightarrow \mathbf{z}^{(3)} \rightarrow}_{\text{initial samples typically garbage}} \ldots \rightarrow \underbrace{\mathbf{z}^{(L-2)} \rightarrow \mathbf{z}^{(L-1)} \rightarrow \mathbf{z}^{(L)}}_{\text{after convergence, actual samples from } p(\mathbf{z})}$$

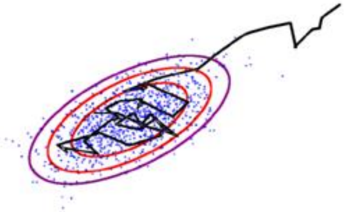


- Given current sample $\boldsymbol{z}^{(\ell)}$ from the chain, MCMC generates the next sample $\boldsymbol{z}^{(\ell+1)}$ as
  - Use a proposal distribution $q(\boldsymbol{z}|\boldsymbol{z}^{(\ell)})$ to generate a candidate sample $\boldsymbol{z}_*$
  - Accept/reject $\boldsymbol{z}_*$ as the next sample based on an acceptance criterion (will see later)
  - If accepted, set $\boldsymbol{z}^{(\ell+1)} = \boldsymbol{z}_*$. If rejected, set $\boldsymbol{z}^{(\ell+1)} = \boldsymbol{z}^{(\ell)}$

Should also have the same support as $p(\boldsymbol{z})$

- Important: The proposal distribution $q(\boldsymbol{z}|\boldsymbol{z}^{(\ell)})$ depends on the previous sample $\boldsymbol{z}^{(\ell)}$

MCMC: The Basic Scheme



- The chain run infinitely long (i.e., upon convergence) will give ONE sample from $p(\boldsymbol{z})$
- But we usually require several samples to approximate $p(\boldsymbol{z})$

MCMC is exact in theory but approximate in practice since we can't run the chain for infinitely long in practice

Thus we say that the samples are approximately from the target distribution

- This is done as follows
  - Start the chain at an initial $\boldsymbol{z}^{(0)}$
  - Using the proposal $q(\boldsymbol{z}|\boldsymbol{z}^{(\ell)})$, run the chain long enough, say T_1 steps
  - Discard the first $T_1 - 1$ samples (called "burn-in" samples) and take last sample $\boldsymbol{z}^{(T_1)}$
  - Continue from $\boldsymbol{z}^{(T_1)}$ up to T_2 steps, discard intermediate samples, take last sample $\boldsymbol{z}^{(T_2)}$
    - This discarding (called "thinning") helps ensure that $\boldsymbol{z}^{(T_1)}$ and $\boldsymbol{z}^{(T_2)}$ are uncorrelated
  - Repeat the same for a total of S times
  - In the end, we now have S *approximately independent* samples from $p(\boldsymbol{z})$

Will treat it as our first sample from $p(\boldsymbol{z})$

Requirement for Monte Carlo approximation

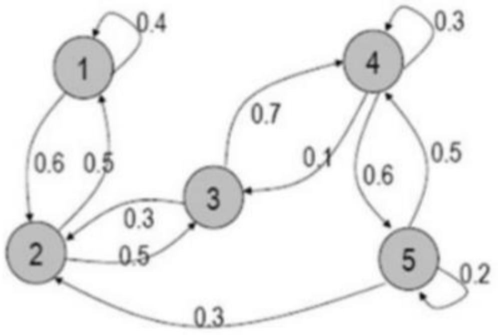
- Note: Good choices for T_1 and $T_i - T_{i-1}$ (thinning gap) are usually based on heuristics

MCMC: Some Basic Theory

- A first order Markov Chain assumes $p\left(\boldsymbol{z}^{(\ell+1)} | \boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(\ell)}\right) = p(\boldsymbol{z}^{(\ell+1)} | \boldsymbol{z}^{(\ell)})$
- A 1st order Markov Chain $\boldsymbol{z}^{(0)}, \boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(L)}$ is a sequence of r.v.'s and is defined by
  - An initial state distribution $p(\boldsymbol{z}^{(0)})$
  - A Transition Function (TF): $T_\ell\left(\boldsymbol{z}^{(\ell)} \to \boldsymbol{z}^{(\ell+1)}\right) = p(\boldsymbol{z}^{(\ell+1)} | \boldsymbol{z}^{(\ell)})$
- TF is a distribution over the values of next state given the value of the current state
- Assuming a K-dim discrete state-space, TF will be $K \times K$ probability table

Transition probabilities can be defined using a *K*x*K* table if **z** is a discrete r.v. with *K* possible values

$$
\begin{array}{c c}
 & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} &
\begin{bmatrix}
0.4 & 0.6 & 0.0 & 0.0 & 0.0 \\
0.5 & 0.0 & 0.5 & 0.0 & 0.0 \\
0.0 & 0.3 & 0.0 & 0.7 & 0.0 \\
0.0 & 0.0 & 0.1 & 0.3 & 0.6 \\
0.0 & 0.3 & 0.0 & 0.5 & 0.2
\end{bmatrix}
\end{array}
$$



- Homogeneous Markov Chain: The TF is the same for all ℓ , i.e., $T_\ell = T$

MCMC: Some Basic Theory

- Consider the following Markov Chain to sample a discrete r.v. $\boldsymbol{z}$ with 3 possible values

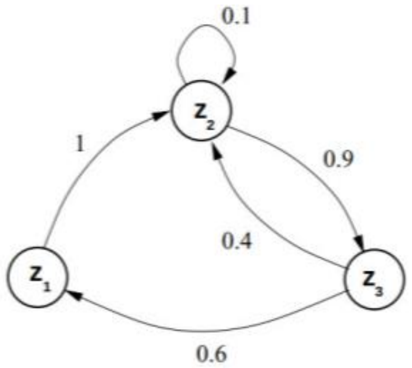
The initial state distribution for $\boldsymbol{z}$

$$p(\boldsymbol{z}^{(0)}) = p\left(z_1^{(0)}, z_2^{(0)}, z_3^{(0)}\right)$$

Probabilities of the initial state taking each of the 3 possible values

$$= [0.5, 0.2, 0.3]$$

$$T = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0.1 & 0.9 \\ 0.6 & 0.4 & 0 \end{bmatrix}$$



Distribution of $\boldsymbol{z}$ after taking the first step

$$p(\boldsymbol{z}^{(1)}) = p(\boldsymbol{z}^{(0)}) \times T = [0.2, 0.6, 0.2] \quad \text{(rounded to single digit after decimal)}$$

After doing it a few more (say some m) times

Stationary/Invariant Distribution $p(\boldsymbol{z})$ of this Markov Chain

$p(\boldsymbol{z})$ is multinoulli with $\pi = [0.2, 0.4, 0.4]$

$$p(\boldsymbol{z}^{(0)}) \times T^m = [0.2, 0.4, 0.4] \qquad \text{(rounded to single digit after decimal)}$$

- $p(\boldsymbol{z})$ being Stationary means no matter what $p(\boldsymbol{z}^{(0)})$ is, we will reach $p(\boldsymbol{z})$
- A Markov Chain has a stationary distribution if T has the following properties
  - Irreducibility: T's graph is connected (ensures reachability from anywhere to anywhere)
  - Aperiodicity: T's graph has no cycles (ensures that the chain isn't trapped in cycles)

MCMC: Some Basic Theory

- A Markov Chain with transition function $\boldsymbol{T}$ has stationary distribution $\boldsymbol{p(z)}$ if $\boldsymbol{T}$ satisfies

Known as the Detailed Balance condition

$$p(\boldsymbol{z})T(\boldsymbol{z}'|\boldsymbol{z}) = p(\boldsymbol{z}')T(\boldsymbol{z}|\boldsymbol{z}')$$

Here $T(b|a)$ denotes the transition probability of going from state $\boldsymbol{a}$ to state $\boldsymbol{b}$

- Integrating out (or summing over) detailed balanced condition on both sides w.r.t. $\boldsymbol{z}'$

Thus $p(z)$ is the stationary distribution of this Markov Chain

$$p(\boldsymbol{z}) = \int p(\boldsymbol{z}')T(\boldsymbol{z}|\boldsymbol{z}')d\boldsymbol{z}'$$

- Thus a Markov Chain with detailed balance always converges to a stationary distribution
- Detailed Balance ensures reversibility
- Detailed balance is sufficient but not necessary condition for having a stationary distr.

Some MCMC Algorithms

Metropolis-Hastings (MH) Sampling (1960)

- Suppose we wish to generate samples from a target distribution $\boldsymbol{p(z)} = \frac{\tilde{p}(\boldsymbol{z})}{Z_p}$
- Assume a suitable proposal distribution $\boldsymbol{q(z|z^{(\tau)})}$, e.g., $\mathcal{N}(\boldsymbol{z}|\boldsymbol{z}^{(\tau)}, \sigma^2 \boldsymbol{I})$
- In each step, draw $\boldsymbol{z}^*$ from $\boldsymbol{q(z|z^{(\tau)})}$ and accept $\boldsymbol{z}^*$ with probability

Favors acceptance of $\boldsymbol{z}^*$ if it is more probable than $\boldsymbol{z}^{(\tau)}$ (under $\boldsymbol{p(z)}$)

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

Downweight the probability of acceptance of z^* if the proposal itself favors its generation (i.e., if $q(z^*|z^{(\tau)})$ is high

- Transition function of this Markov Chain
  - $T\left(\boldsymbol{z}^*|\boldsymbol{z}^{(\tau)}\right) = A(\boldsymbol{z}^*, \boldsymbol{z}^{(\tau)})q(\boldsymbol{z}^*|\boldsymbol{z}^{(\tau)})$ if state changed
  - $T\left(\boldsymbol{z}^*|\boldsymbol{z}^{(\tau)}\right) = q\left(\boldsymbol{z}^{(\tau)}|\boldsymbol{z}^{(\tau)}\right) + \sum_{\boldsymbol{z}^* \neq \boldsymbol{z}^{(\tau)}}(1 - A(\boldsymbol{z}^*, \boldsymbol{z}^{(\tau)}))q(\boldsymbol{z}^*|\boldsymbol{z}^{(\tau)})$ otherwise

Can Show that this TF satisfied detailed balance

The MH Sampling Algorithm

- Initialize $\boldsymbol{z}^{(1)}$ randomly
- For $\ell = 1,2,\ldots,L$
  - Sample $\boldsymbol{z}^* \sim q(\boldsymbol{z}^*|\boldsymbol{z}^{(\ell)})$ and $u \sim \mathrm{Unif}(0,1)$
  - 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 $A\left(\boldsymbol{z}^*, \boldsymbol{z}^{(\ell)}\right) > u$

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

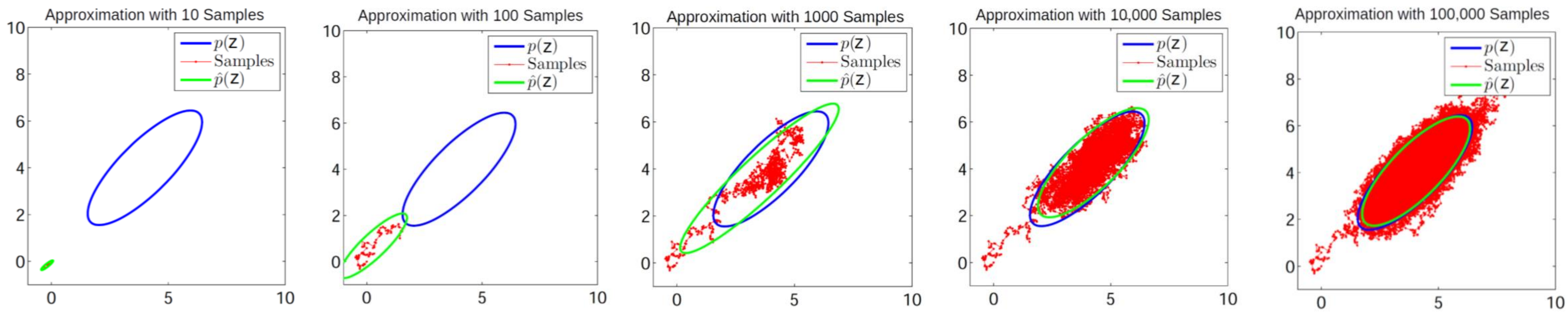
Meaning accepting $\boldsymbol{z}^*$ with probability $A\left(\boldsymbol{z}^*, \boldsymbol{z}^{(\ell)}\right)$

  - Else

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

MH Sampling in Action: A Toy Example..

- Target distribution $p(\boldsymbol{z}) = \mathcal{N}\left(\begin{bmatrix}4\\4\end{bmatrix}, \begin{bmatrix}1 & 0.8\\0.8 & 1\end{bmatrix}\right)$
- Proposal distribution $q(\boldsymbol{z}^{(t)}|\boldsymbol{z}^{(t-1)}) = \mathcal{N}\left(\boldsymbol{z}^{(t-1)}, \begin{bmatrix}0.01 & 0\\0 & 0.01\end{bmatrix}\right)$

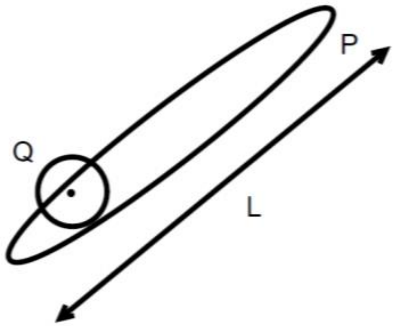


MH Sampling: Some Comments

- 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")



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

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 $\boldsymbol{p(z)} = \frac{\tilde{p}(\boldsymbol{z})}{Z_p}$ is some target posterior then $\tilde{\boldsymbol{p}}(\boldsymbol{z})$ 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 $\boldsymbol{z} = [z_1, z_2, \ldots, z_M]$
- 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

- M: Total number of variables, T: number of Gibbs sampling iterations

1. Initialize $\{z_i : i = 1, \ldots, M\}$

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

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

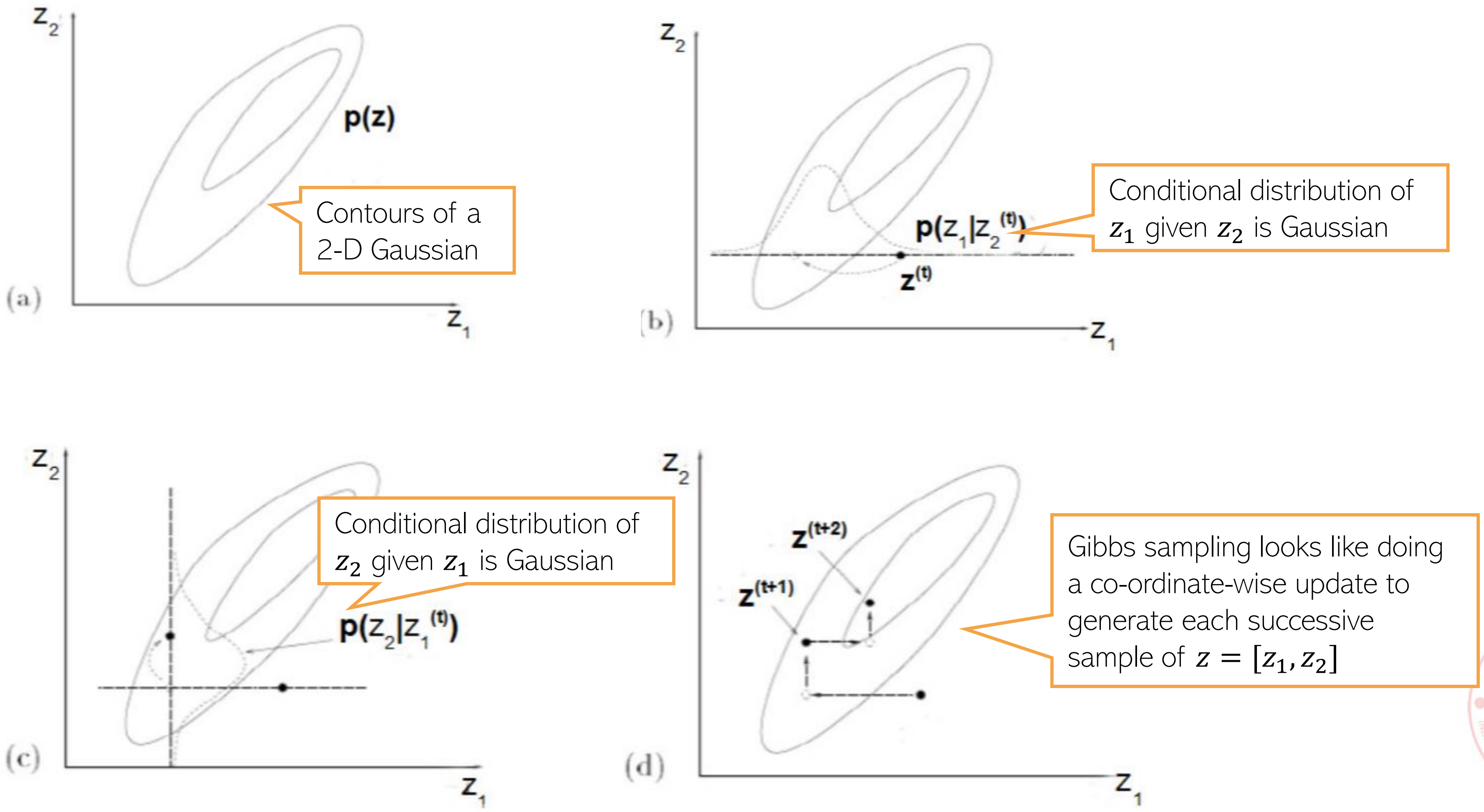
CP of each component of $\boldsymbol{z}$ uses the most recent values (from this or the previous iteration) of all the other components

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: Another Simple Example

- Bayesian linear regression: $p(y_n|\boldsymbol{x}_n, \boldsymbol{w}, \beta) = \mathcal{N}(y_n|\boldsymbol{w}^\top \boldsymbol{x}_n, \beta^{-1})$, $p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|0, \lambda^{-1}I)$, $p(\lambda) = \text{Gamma}(\lambda|a, b)$, $p(\beta) = \text{Gamma}(\beta|c, d)$. Gibbs sampler for $p(w, \lambda, \beta|\boldsymbol{X}, \boldsymbol{y})$ will be
- Initialize λ, β as $\lambda^{(0)}, \beta^{(0)}$. For iteration $t = 1,2, \dots, T$
  - Generate a random sample of $\boldsymbol{w}$ by sampling from its CP as

$$\boldsymbol{w}^{(t)} \sim \mathcal{N}(\boldsymbol{w}|\boldsymbol{\mu}^{(t-1)}, \boldsymbol{\Sigma}^{(t-1)})$$

where

$$\boldsymbol{\Sigma}^{(t-1)} = \left(\beta^{(t-1)}\boldsymbol{X}^\top \boldsymbol{X} + \lambda^{(t-1)}\right)^{-1}$$

$$\boldsymbol{\mu}^{(t-1)} = \left(\boldsymbol{X}^\top \boldsymbol{X} + \frac{\lambda^{(t-1)}}{\beta^{(t-1)}}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y}$$

  - Generate a random sample of λ by sampling from its CP as

$$\lambda^{(t)} \sim \text{Gamma}\left(\lambda|a + \frac{D}{2}, b + \frac{\boldsymbol{w}^{(t)^\top} \boldsymbol{w}^{(t)}}{2}\right)$$

  - Generate a random sample of β by sampling from its CP as

$$\beta^{(t)} \sim \text{Gamma}\left(\beta|c + \frac{N}{2}, d + \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}^{(t)}\|^2}{2}\right)$$

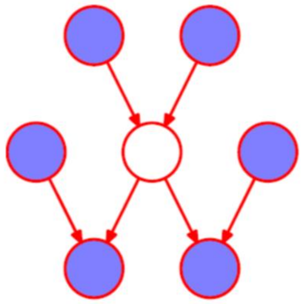
- The posterior's approximation is the set of collected samples

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

Deriving A Gibbs Sampler: The General Recipe

- Suppose the target is an intractable posterior $p(\boldsymbol{Z}|\boldsymbol{X})$ where $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_M]$
- 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)



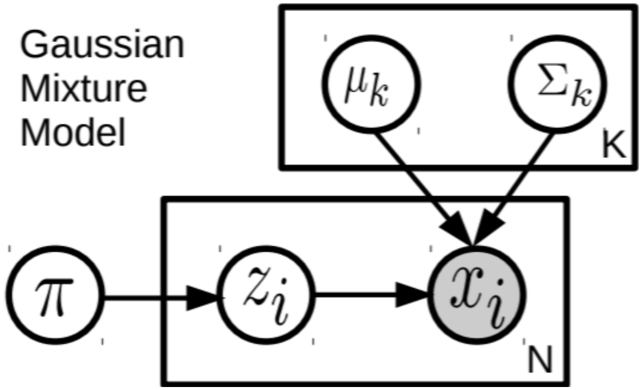
Markov Blanket

- 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

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

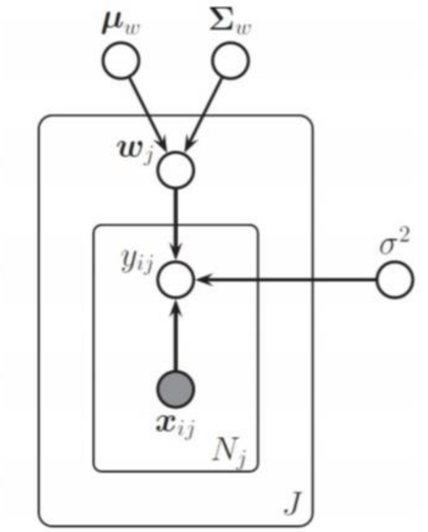
$$
p(z_i = k|\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) \propto \pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)
$$

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

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

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

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

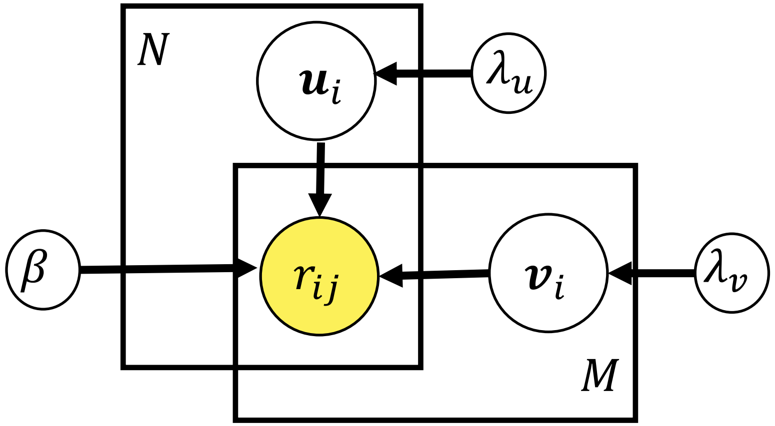
$$\begin{aligned} p(\mathbf{w}_j|\mathcal{D}_j, \boldsymbol{\theta}) &= \mathcal{N}(\mathbf{w}_j|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \\ \boldsymbol{\Sigma}_j^{-1} &= \boldsymbol{\Sigma}^{-1} + \mathbf{X}_j^T \mathbf{X}_j/\sigma^2 \\ \boldsymbol{\mu}_j &= \boldsymbol{\Sigma}_j(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}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

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

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

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

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

$$p(\boldsymbol{u}_i|\mathbf{R},\mathbf{V}) = \mathcal{N}(\boldsymbol{u}_i|\boldsymbol{\mu}_{u_i}, \boldsymbol{\Sigma}_{u_i})$$

$$\boldsymbol{\Sigma}_{u_i} = (\lambda_u \mathbf{I} + \beta \sum_{j:(i,j)\in\Omega} \mathbf{v}_j \mathbf{v}_j^\top)^{-1}$$

$$\boldsymbol{\mu}_{u_i} = \boldsymbol{\Sigma}_{u_i}(\beta \sum_{j:(i,j)\in\Omega} r_{ij}\mathbf{v}_j)$$

$$p(\boldsymbol{v}_j|\mathbf{R},\mathbf{U}) = \mathcal{N}(\boldsymbol{v}_j|\boldsymbol{\mu}_{v_j}, \boldsymbol{\Sigma}_{v_j})$$

$$\boldsymbol{\Sigma}_{v_j} = (\lambda_v \mathbf{I} + \beta \sum_{i:(i,j)\in\Omega} \boldsymbol{u}_i \boldsymbol{u}_i^\top)^{-1}$$

$$\boldsymbol{\mu}_{v_j} = \boldsymbol{\Sigma}_{v_j}(\beta \sum_{i:(i,j)\in\Omega} r_{ij}\boldsymbol{u}_i)$$

$$p(\lambda_u|\mathbf{U}) = \text{Gamma}(\lambda_u|a + 0.5 * NK, b + 0.5 * \sum_{i=1}^N \boldsymbol{u}_i^\top \boldsymbol{u}_i)$$

$$p(\lambda_v|\mathbf{V}) = \text{Gamma}(\lambda_v|c + 0.5 * MK, d + 0.5 * \sum_{j=1}^M \boldsymbol{v}_j^\top \boldsymbol{v}_j)$$

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

$\boldsymbol{\Omega}$ denotes the indices that are observed in the ratings matrix

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

Using MCMC samples to make predictions

- Using the S samples $\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)}, \dots, \boldsymbol{Z}^{(S)}$, 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