10-708: Probabilistic Graphical Models 10-708, Spring 2016

17 : MCMC (cont'd) and Intro to Topic Modeling

Lecturer: Matthew Gormley *Scribes: Chun-Liang Li, Yanyu Liang, Mengxin Li*

1 MCMC with Auxiliary Variables

In Gibbs sampling, we try to sample "less" variables in each time. Therefore, we only sample one variable by conditioning on the remaining ones. However, we could do a reverser way by introducing more variables to be sampled.

For any distribution $p(x)$, we know that $p(x) = \int_u p(x,u)$. If we want to sample from certain distribution $p(x)$, which is hard to sample. We then introduce the "auxiliary" variable u, which is a random variable that do not exist in the model but are introduced into the model to facilitate sampling. We then hope the joint distribution $p(x,u)$ is easy to navigate, and the conditional distributions $p(x|u)$ and $p(u|x)$ are easy to sample. Then sampling from $p(x,u)$ is easier than sampling from $p(x)$, and we can marginalize out u to get $p(x)$ back.

Next, we discuss two approaches, including slice sampling and Hamiltonian Monte Carlo.

1.1 Slice Sampling

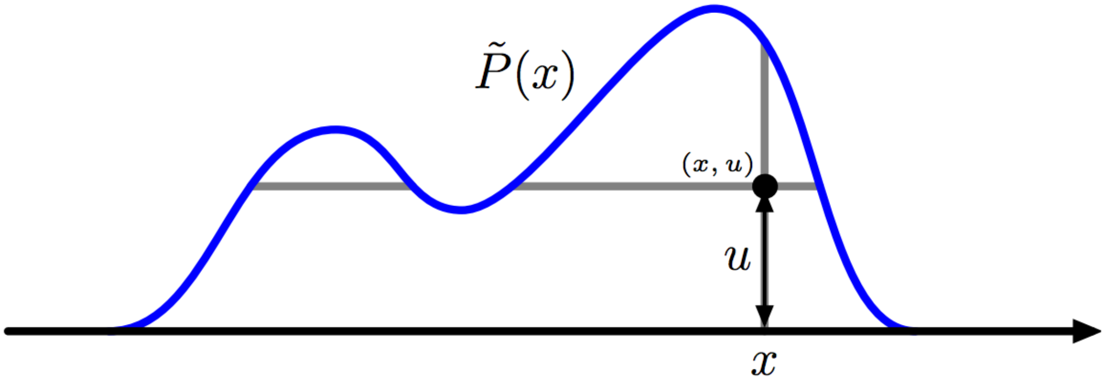

Figure 1: Slice Sampling.

Assume we want to sample from $P(x)$, and $\tilde{P}(x) \propto P(x)$, where we can evaluate $\tilde{P}(x)$. The we define u as the auxiliary variable, and $p(u|x)$ is the uniform sampling between 0 and $\tilde{P}(x)$. Also, we define $p(x|u)$ as the uniform sampling from $\{x'|\tilde{P}(x') \geq u\}$. The sample (x,u) are uniformly distributed under the area of $\tilde{P}(x)$. We can then obtain $p(x)$ by marginalizing out u.

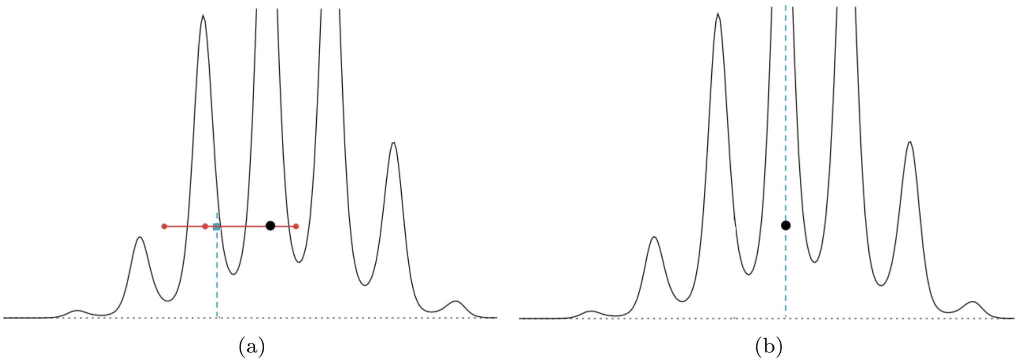

Figure 2: Example of the sampling procedure of slice sampling.

The algorithm is shown as follows and Figure 1.1.

- Initialize x'.
- Sample $u \sim \text{Unif}\,(0, \tilde{p}(x'))$.
- Sample x uniformly from $\{z|\tilde{P}(z) \geq u\}$.

1.1.1 Computational Concern

Finding the set $z|\tilde{P}(z) > u$ can be computationally expensive or unfeasible. Then bracket slice sampling can be applied. That is we use a horizon bracket to contain x', then extend or shrink the bracket to search the proper size of bracket that is $z|\tilde{P}(z) > u$.

1.1.2 Discussion

The advantages of slice sampling includes

- Without tuning parameters
- No rejection.

However, this method is still suffered from random walk as Gibbs sampling.

1.2 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo is another auxiliary variable method, which makes use of not only the probability distribution but the gradient of probability for sampling. Consider a Boltzmann distribution of $\boldsymbol{x} \in \mathbb{R}^N$, it can be wiritten as $p(\boldsymbol{x}) = Z_x^{-1}\exp\{-E(\boldsymbol{x})\}$. By introducing an independent auxiliary variable $\boldsymbol{q} \in \mathbb{R}^N, \boldsymbol{q} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, the joint distribution is $p(\boldsymbol{x}, \boldsymbol{q}) = Z_x^{-1}Z_q^{-1}\exp\{-E(\boldsymbol{x}) - \boldsymbol{q}^T\boldsymbol{q}/2\}$. Then the sampling is divided into two steps. The first step is similar to block Gibbs sampling, namely given $\boldsymbol{x}^{(t)}$ sample $\boldsymbol{q}^{(t+1)}$. And the second step is to update $\boldsymbol{x}$. Since $\boldsymbol{x}$ and $\boldsymbol{q}$ are independent, $p(\boldsymbol{q}|\boldsymbol{x})$ is Gaussian and to sample from it is simple. The update of $\boldsymbol{x}$ follows Metropolis algorithm, and more specifically, the transition is proposed by Hamiltonian Dynamics. The following will first introduce Hamiltonian Dynamics and its properties, and then talks about Hamiltonian Monte Carlo in details.

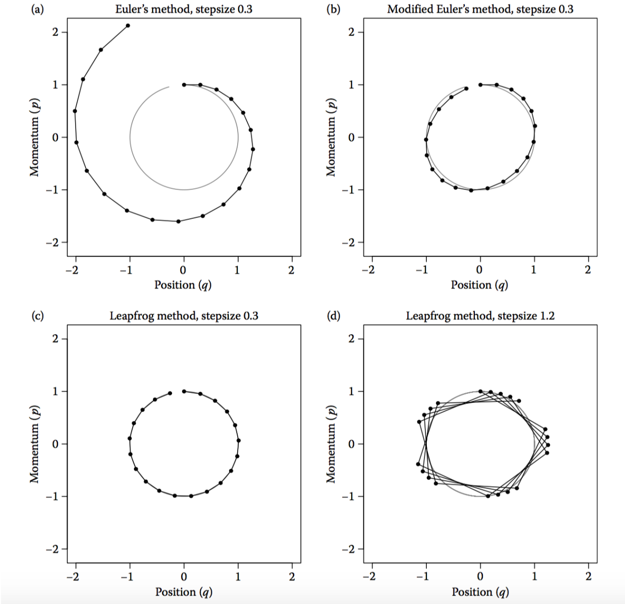

Figure 3: Euler's method and Leapfrog algorithm for approximating Hamiltonian Dynamics

1.2.1 Hamiltonian Dynamics

Given two physical quantities $\boldsymbol{x}, \boldsymbol{q} \in \mathbb{R}^N$ and let $H(\boldsymbol{x}, \boldsymbol{q})$ be the Hamiltonian of the system, Hamiltonian Dynamics describes the evolution of them over time, which follows the following relationships [1]:

$$\frac{dx_i}{dt} = \frac{\partial H}{\partial q_i} \tag{1}$$

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial x_i} \tag{2}$$

A simple example is to consider a simple harmonic motion in one dimension, where x is the position and q is the momentum and let the Hamiltonian be $-kx + q^2/2m$. From Eqs. (1) to (2), we get:

$$\frac{dq}{dt} = -k \quad \frac{dx}{dt} = \frac{q}{m}$$

From Newton's Law, the mechanical energy (namely the Hamiltonian) of the above motion is unchanged. And it turns out that Hamiltonian Dynamic always gaurantees that the Hamiltonian is invariant over time. Additionally, the evolution over time is reversible in Hamiltonian Dynamics. Namely, if from $\boldsymbol{x}, \boldsymbol{q}$, the evolution goes Δt time and achieves $\boldsymbol{x}', \boldsymbol{q}'$, it will take exactly the same time for the system to evolve from $\boldsymbol{x}', -\boldsymbol{q}'$ to $\boldsymbol{x}, -\boldsymbol{q}$.

By discretizing the time, Hamiltonian Dynamics can be simulated by computer. Euler's method and Leapfrog alogrithm are two ways to do so, and their update rules are slightly different which give different performances. 3 [1] shows the simulation of system with $x^2/2 + q^2/2$ as Hamiltonian. Sometimes Euler's method does not converge and the accuracy is low. By contrast, Leapfrog is more stable and still satisfies reversibility. The update rule of Leapfrog method is showed as follow:

$$
\begin{aligned}
q_i(t+\epsilon/2) &= q_i(t) - (\epsilon/2)\left.\frac{\partial E}{\partial x_i}\right|_t \\
x_i(t+\epsilon) &= x_i(t) + \epsilon\frac{q_i(t+\epsilon/2)}{m_i} \\
q_i(t\epsilon) &= q_i(t+\epsilon/2) - (\epsilon/2)\left.\frac{\partial E}{\partial x_i}\right|_{t+\epsilon}
\end{aligned}
$$

, where ϵ is step size. And in practice, once we fix step size and number of steps, from a starting state $\boldsymbol{x}, \boldsymbol{q}$, Leapfrog method proposes a new state $\boldsymbol{x}', \boldsymbol{q}'$ where $H(\boldsymbol{x}, \boldsymbol{q})$ is approximately equal to $H(\boldsymbol{x}', \boldsymbol{q}')$, and such transition is reversible.

1.2.2 Algorithm and Properties of HMC

In our case, let $\boldsymbol{q}^T\boldsymbol{q}/2 = K(\boldsymbol{q})$ and set the Hamiltonian of the system as $H(\boldsymbol{x}, \boldsymbol{q}) = E(\boldsymbol{x}) + K(\boldsymbol{q})$. The algorithm of Hamiltonian Monte Carlo (HMC) is as follow:

Step 1
Draw new sample $\boldsymbol{q}' \sim p(\boldsymbol{q}|\boldsymbol{x})$

Step 2
Run Leapfrog on $H(\boldsymbol{x}, \boldsymbol{q})$ with step size ϵ for L steps
And obtain $\boldsymbol{x}', \boldsymbol{q}''$, where $H(\boldsymbol{x}, \boldsymbol{q}') \simeq H(\boldsymbol{x}', \boldsymbol{q}''$

Step 3
Use $\boldsymbol{x}', \boldsymbol{q}''$ as proposed sample, and accept with probability:
$\min\{1, \exp\{H(\boldsymbol{x}, \boldsymbol{q}') - H(\boldsymbol{x}', \boldsymbol{q}'')\}\}$

Step 1 is block Gibbs sampling step, where the proposal distribution is a conditional distribution. By constraction, $p(\boldsymbol{q}|\boldsymbol{x}) = p(\boldsymbol{q}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, so it satisifies detailed balance. Step 2 and 3 are Metropolis algorithm. The transition from $\boldsymbol{x}, \boldsymbol{q}'$ to $\boldsymbol{x}', \boldsymbol{q}''$ is defined by Hamiltonian Dynamics. Since Leapfrog method is reversible, $R(\boldsymbol{x}', \boldsymbol{q}'' \leftarrow \boldsymbol{x}, \boldsymbol{q}') = R(\boldsymbol{x}, -\boldsymbol{q}' \leftarrow \boldsymbol{x}', -\boldsymbol{q}'') = 1$. If we artificially reverse the direction of $\boldsymbol{q}''$ to $-\boldsymbol{q}''$ after running Leapfrog, we have $R(\boldsymbol{x}', -\boldsymbol{q}'' \leftarrow \boldsymbol{x}, \boldsymbol{q}') = R(\boldsymbol{x}, \boldsymbol{q}' \leftarrow \boldsymbol{x}', -\boldsymbol{q}'') = 1$, which stastifies Metropolis algorithm's condition. Using the fact that $K(\boldsymbol{q}) = K(-\boldsymbol{q})$, the acceptance rule can be simplified as:

$$
\begin{aligned}
\because H(\boldsymbol{x}', -\boldsymbol{q}'') &= H(\boldsymbol{x}', \boldsymbol{q}'') \\
H(\boldsymbol{x}, -\boldsymbol{q}') &= H(\boldsymbol{x}, \boldsymbol{q}') \\
\therefore \min\{1, \frac{p(\boldsymbol{x}', -\boldsymbol{q}'')}{p(\boldsymbol{x}, -\boldsymbol{q}')}\} &= \min\{1, \exp\{H(\boldsymbol{x}, -\boldsymbol{q}') - H(\boldsymbol{x}', -\boldsymbol{q}'')\}\} \\
&= \min\{1, \exp\{H(\boldsymbol{x}, \boldsymbol{q}') - H(\boldsymbol{x}', \boldsymbol{q}'')\}\}
\end{aligned}
$$

, which is exactly what checked in Step 3. Metropolis algorithm satisfies detailed balance, so HMC satisfies detailed balance.

Ideally, Hamiltonian Dynamics has invariant H, which leads to 100% acceptance. But, in reality, Leapfrog method may introduce some errors, so Step 3 is necessary, but the acceptance rate is very high. Fig. 4 [1]

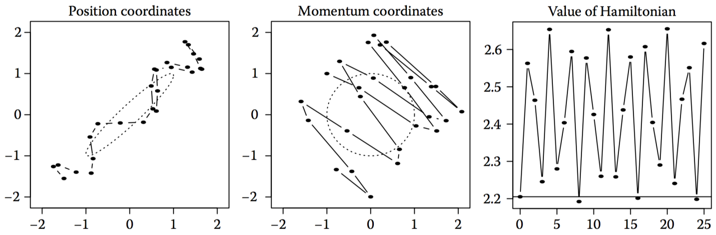

Figure 4: The trajectory for a two-dimensional Gaussian distribution sampled by HMC

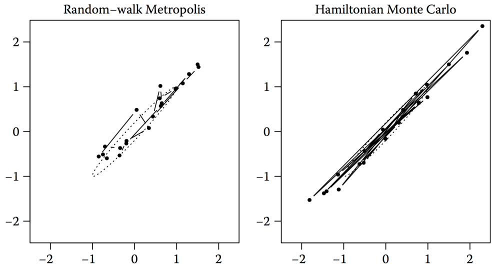

Figure 5: HMC vs. M-H

shows an example of sampling a 2-D Gaunssian using HMC. Unlike random walk, the trajectory gose from lower left-hand side to upper right-hand side because of the Leapfrog step. Fig. 5 [1] is the comparison between HMC and M-H algorithm's results. Thanks to Hamiltonian Dynamic, sample is more likely to make big jump across the space but still holds a high probability to be accepted.

2 Introduction to Topic Modeling

Topic Modeling is a method (ususally unsupervised) for discovery of latent or hidden structure in a corpus. Suppose you're given a massive corpora and asked to carry out the following tasks:

- Organize the documents into thematic categories.
- Describe the evolution of those categories over time.
- Enable a domain expert to analyze and understand the content.
- Find relationships between the categories.
- Understand how authorship influences the content.

Topic modeling provides a modeling toolbox for these tasks. Although it is applied primarily to text corpora, the techniques can be generalized to solve problems in other fields including computer vision and bioinformatics.

2.1 Beta Bernoulli Model

Beta Bernoulli Model is a simple Bayesian model that can be used to model corpus in which the words are binary random variables whose prior is modeled by the Beta distribution. Beta distribution is a conjugate distribution that can be written as:

$$f(\phi|\alpha,\beta) = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

The Beta distribution is illustrated in Figure 2.1.

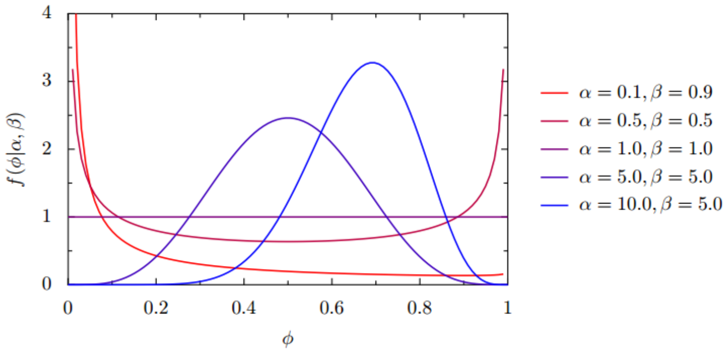

Figure 6: Beta distribution

The generative process for Beta Bernoulli model can be described as:

- draw $\phi \sim Beta(\alpha, \beta)$
- For each word $n \in \{1, ..., N\}$:
  draw $x_n \sim Bernoulli(\phi)$

2.2 Dirichlet-Multinomial Model

Similar to Beta Bernoulli Model, the Dirichlet-Multinomial Model can also model corpus but in which the words are multinomial random variables whose prior is modeled by Dirichlet distribution. Dirichlet distribution is a conjugate distribution that can be written as:

$$p(\vec{\phi}|\alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^{K} \phi_k^{\alpha_k - 1}$$

Where

$$B(\alpha) = \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k - 1)}$$

The Dirichlet distribution is illustrated in Figure 2.2.

The generative process for Dirichlet-multinomial model can be described as:

- draw $\phi \sim Dir(\beta)$
- For each word $n \in \{1, ..., N\}$:
  Draw $x_n \sim Mult(1, \phi)$

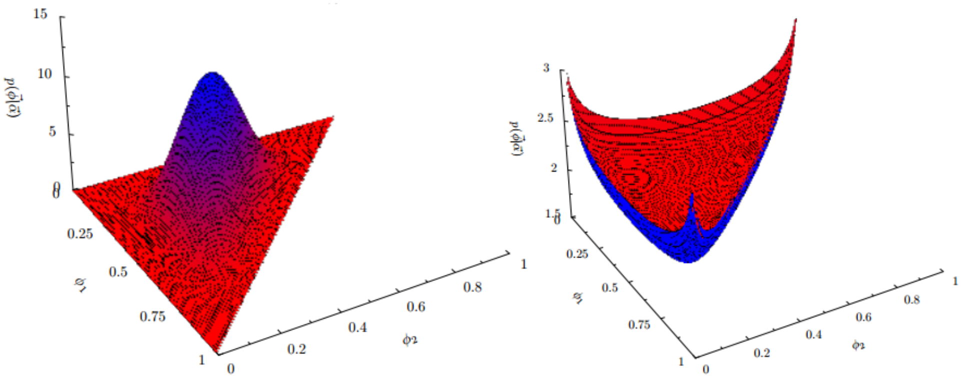

Figure 7: Dirichlet distribution

2.3 Dirichlet-Multinomial Mixture Model

When we take one step further from the Dirichlet-multinomial model, instead of just generating words, we also want to generate independent documents so that each document has a particular topic. The generative process for Dirichlet-multinomial mixture model can be described as below:

- For each topic $k \in \{1, ..., K\}$:

  draw $\phi_k \sim Dir(\beta)$

- Draw $\theta \sim Dir(\alpha)$

- For each document $m \in \{1, ..., M\}$:

  Draw $z_m \sim Mult(1, \theta)$

  For each word $n \in \{1, ..., N_m\}$, draw $x_n \sim Mult(1, \phi_{z_m})$

2.4 Latent Dirichlet Allocation

The problem of Dirichlet-multinomail model is that it cannot model documents which have more than one topic. The Latent Dirichlet Allocation (LDA) is proposed to tackle this drawback of Dirichlet-multinomial model. In LDA, for each document, there is a probability distribution over the topics. The generative process for LDA can be described as below:

- For each topic $k \in \{1, ..., K\}$:

  draw $\phi_k \sim Dir(\beta)$

- For each document $m \in \{1, ..., M\}$:

  Draw $\theta_m \sim Dir(\alpha)$

  For each word $n \in \{1, ..., N_m\}$:

    Draw $z_{mn} \sim Mult(1, \theta_m)$

    Draw $x_{mn} \sim Mult(1, \phi_{z_{mi}})$

References

[1] R. M. Neal *et al.*, "Mcmc using hamiltonian dynamics," *Handbook of Markov Chain Monte Carlo*, vol. 2, pp. 113–162, 2011.