Sampling Tutorial

Taha Bahadori

University of Southern California

November 4, 2012

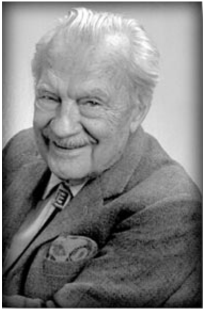
Outline

- Introduction
- The Sampling Problems
- Random Walk Based Algorithms:
    - Metropolis-Hastings Algorithm
    - Gibbs Sampling
    - Convergence Diagnosis
    - Auxiliary Variable Methods
- Hybrid Monte Carlo (HMC)
- Sequential Monte Carlo (SMC) esp. Particle Filtering
- Others

History

$$\text{Metropolis} \xrightarrow{\text{Generalized}} \text{Metropolis Hastings} \xrightarrow{\text{Special Case}} \text{Gibbs Sampling}$$

- All developments are done in Computational Physics
- The Landmark 1953 Paper
  N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, *Equation of state calculations by fast computing machines,* Journal of Chemical Physics.
- Metropolis was the supervisor in Los Alamos National Lab.
  "Metropolis played no role in its development other than providing computer time!"



Perquisite: Markov Chains

The random process $X_t \in \mathcal{S}$, for $t = 1, \ldots, T$ has Markov property iff:

$$p(X_t | X_{t-1}, X_{t-2}, \ldots, X_1) = p(X_t | X_{t-1}).$$

Finite-state Discrete Time Markov Chains $|\mathcal{S}| < \infty$ can be completely specified by the transition matrix.
The transition matrix P defined by the elements

$$P = [p_{ij}]; \quad p_{ij} = \mathbb{P}[X_t = j | X_{t-1} = i].$$

For *Irreducible* chains, the stationary distribution π is long-term proportion of time that the chain spends in each state. Computed by $\pi = \pi P$.

Note: In order to understand the proof the "Time Reversibility" and "Detailed Balanced" concepts are required.

Problem Definition

The goal is to compute the following expectation:

$$\mathbb{E}[f] = \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

Sampling: obtain a set of samples $\{\mathbf{z}^{(i)}\}$ where $i = 1, \ldots, N$ drawn independently from $p(\mathbf{z})$ and approximate the expectation as:

$$\mathbb{E}[f] \approx \hat{\mathbb{E}}[f] = \frac{1}{N}\sum_{i=1}^{N} f\left(\mathbf{z}^{(i)}\right), \quad \mathbf{z}^{(i)} \sim p(\mathbf{z}).$$

Criteria for a Good Sampling Algorithm:
With the smallest N, gives the best approximation of $\mathbb{E}[f]$.
Example:
Find the average height of Americans

Significance

Bayesian Inference

$$p(\mathbf{w}|\mathbf{X}) \propto \int p(\mathbf{X}|\mathbf{w})p(\mathbf{w})d\mathbf{w}$$

EM Algorithm, The expectation step:

$$Q\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}\right) = \int \ln p(\mathbf{Z}, \mathbf{X}|\boldsymbol{\theta})p\left(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{old}\right) d\mathbf{Z}$$

Approximation Effects
Bayesian Inference The estimation is unbiased and the variance vanishes with rate proportional to $\frac{1}{N}$.
EM Algorithm: Generalized EM guarantees the convergence despite approximation.

The Basic Methods

$$x \sim p(x)$$

Draw $u \sim Unif(0,1)$

$$x = F_x^{-1}(u),$$

where $F_x(x)$ is the CDF of x

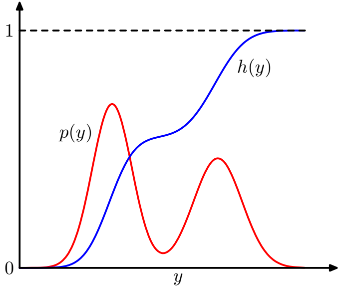


Figure Credit: Chris Bishop *PRML* 2006.

What if we cannot calculate the CDF in closed form?

$$p(\mathbf{x}) = \frac{1}{Z}\tilde{p}(\mathbf{x})$$

Source: Probabilistic Graphical Models

Rejection Sampling

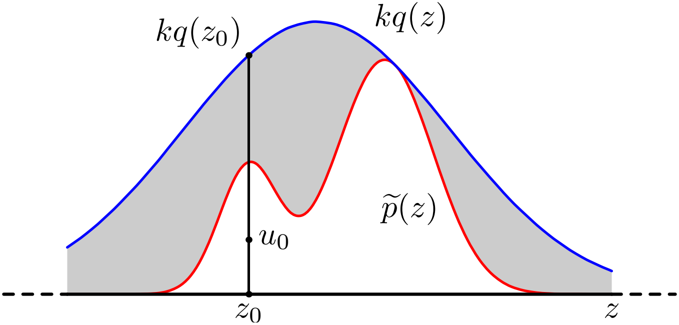


Figure Credit: Chris Bishop *PRML* 2006.

Question: What is the average acceptance ratio?

Metropolis-Hastings (I): The Main Idea

Cannot sample directly from the target distribution?

$\Rightarrow$ Create a Markov chain whose transition matrix does not depend on the normalization term.

$\Rightarrow$ Make sure the chain has a stationary distribution and it is equal to the target distribution.

$\Rightarrow$ After sufficient number of iterations, the chain will converge the stationary distribution.

The Algorithm

- Propose a move from the current state $q(\mathbf{y}|\mathbf{x}_i)$, e.g. $\mathcal{N}(\mathbf{x}_i, \sigma^2\mathbf{I})$
- Accept with probability $\min\left(\frac{p(\mathbf{y})q(\mathbf{y},\mathbf{x}_i)}{p(\mathbf{x}_i)q(\mathbf{x}_i,\mathbf{y})}, 1\right)$
- Otherwise stay in the current location

Metropolis-Hastings (II): Details

Task: Sample from $p(\mathbf{X})$ with discrete values $\mathbf{X} \in \{\mathbf{x}_j, j \geq 1\}$
Create a Markov chain $\mathbf{X}_n$. We want to make its stationary distribution $\pi(\mathbf{X})$ equal to $p(\mathbf{X})$.
Assume that you are at state $\mathbf{X}_n = \mathbf{x}_i$. Select a proposal function $q(\mathbf{x}_i, \mathbf{x}_j)$. Generate a sample $\mathbf{y}$ with $\mathbb{P}\{\mathbf{Y} = \mathbf{x}_j\} = q(\mathbf{x}_i, \mathbf{x}_j)$. With probability $\alpha(\mathbf{x}_i, \mathbf{x}_j)$ set $\mathbf{X}_{n+1} = \mathbf{y}$ and $1 - \alpha(\mathbf{x}_i, \mathbf{x}_j)$ set $\mathbf{X}_{n+1} = \mathbf{X}_n$
The transition matrix:

$$\begin{array}{ll} P_{ij} = q(\mathbf{x}_i, \mathbf{x}_j)\alpha(\mathbf{x}_i, \mathbf{x}_j) & \text{if } j \neq i; \\ P_{ii} = q(\mathbf{x}_i, \mathbf{x}_i) + \sum_{k \neq i} q(\mathbf{x}_i, \mathbf{x}_k)(1 - \alpha(\mathbf{x}_i, \mathbf{x}_k)) & \text{Otherwise;} \end{array}$$

The chain will be time reversible and have stationary probability $\pi(\mathbf{X})$ if: $\pi(\mathbf{x}_i)P_{ij} = \pi(\mathbf{x}_j)P_{ji}$ for $i \neq j$. Setting

$$\pi(\mathbf{x}_i)q(\mathbf{x}_i, \mathbf{x}_j)\alpha(\mathbf{x}_i, \mathbf{x}_j) = \pi(\mathbf{x}_j)q(\mathbf{x}_j, \mathbf{x}_i)\alpha(\mathbf{x}_j, \mathbf{x}_i) \tag{1}$$

Selecting $\alpha(\mathbf{x}_i, \mathbf{x}_j) = \min\left(\frac{\pi(\mathbf{x}_j)q(\mathbf{x}_j, \mathbf{x}_i)}{\pi(\mathbf{x}_i)q(\mathbf{x}_i, \mathbf{x}_j)}, 1\right)$ satisfies Equation (1).

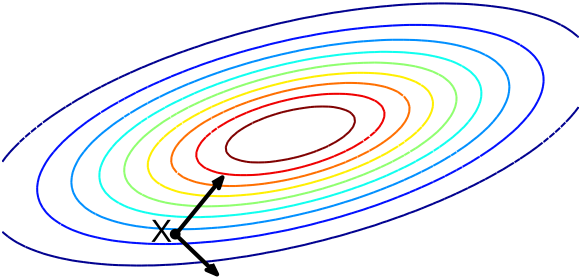
Metropolis-Hastings (III): The Algorithm

To draw N samples from $p(\mathbf{X})$:

Select a proposal distribution $q(\mathbf{x}_2|\mathbf{x}_1)$
Initialize $\mathbf{x}_1$
for $i = 1 \rightarrow MaxIteration$ **do**
    Draw $\mathbf{y} \sim q(\mathbf{y}|\mathbf{x}_i)$
    $\alpha(\mathbf{x}_i, \mathbf{y}) \leftarrow \min\left(\frac{p(\mathbf{y})q(\mathbf{y},\mathbf{x}_i)}{p(\mathbf{x}_i)q(\mathbf{x}_i,\mathbf{y})}, 1\right)$
    Draw $u \sim Unif(0, 1)$
    if $u < \alpha(\mathbf{x}_i, \mathbf{y})$ **then**
        $\mathbf{x}_{i+1} \leftarrow \mathbf{y}$,
    else
        $\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i$
    end if
end for
return Last N samples

Metropolis-Hastings (V): Understanding The Algorithm

Learning From The Past
Task: Sample from $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.
Proposal function: $q(\mathbf{x}_{i+1}|\mathbf{x}_i) = \mathcal{N}(\mathbf{x}_i, \rho\mathbf{I})$



$$\alpha(\mathbf{x}_i, \mathbf{y}) = \min\left(\frac{p(\mathbf{y})q(\mathbf{y},\mathbf{x}_i)}{p(\mathbf{x}_i)q(\mathbf{x}_i,\mathbf{y})}, 1\right) = \min\left(\frac{p(\mathbf{y})}{p(\mathbf{x}_i)}, 1\right)$$

Metropolis-Hastings (VI): Properties (A)

Trade-off between Mixing rate and Acceptance ratio

Definition

$$\text{Acceptance ratio} = \mathbb{E}\left[\alpha(\mathbf{x}_i, \mathbf{y})\right]$$

Mixing rate = the rate that the chain moves around the distribution.

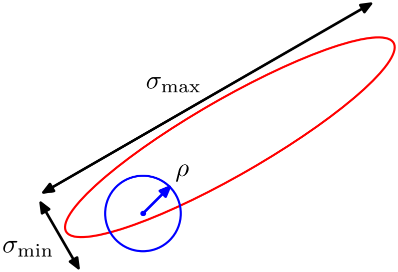


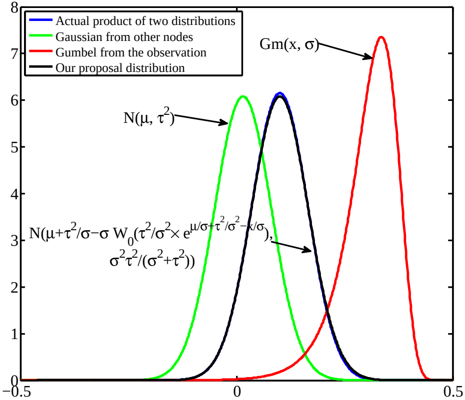
Figure Credit: Chris Bishop *PRML* 2006.

Good Proposal Functions?

Metropolis-Hastings (VI): Good Proposal Functions

Task: Sample from:

$$p(z) \propto \frac{1}{\sqrt{2\pi\tau}} e^{\frac{(z-\mu)^2}{2\tau^2}} \times \frac{1}{\sigma} e^{-\frac{(x-z)}{\sigma}} e^{-e^{-\frac{(x-z)}{\sigma}}}$$



Metropolis-Hastings (VI): Properties (B)

We can have multiple transition matrices P_i (i.e. proposal functions). Under some technical conditions, applying them in turn results in the net transition matrix will be:

$$P^{\star} = \frac{1}{K} \sum_{i=1}^{K} P_i$$

Gibbs Sampling

Task: Sample from the unnormalized joint distribution $\tilde{p}(x_1, \ldots, x_n)$.

Gibbs Sampling

Initialize $x_1, \ldots, x_n$.
for $\tau = 1 \rightarrow MaxIteration$ **do**
    Sample $x_1^{(\tau+1)} \sim p(x_1 | x_2^{(\tau)}, x_3^{(\tau)}, \ldots, x_n^{(\tau)})$.
    Sample $x_2^{(\tau+1)} \sim p(x_2 | x_1^{(\tau+1)}, x_3^{(\tau)}, \ldots, x_n^{(\tau)})$.
    $\vdots$
    Sample $x_j^{(\tau+1)} \sim p(x_j | x_1^{(\tau+1)}, \ldots, x_{j-1}^{(\tau+1)}, x_{j-1}^{(\tau)}, \ldots, \ldots, x_n^{(\tau)})$.
    $\vdots$
    Sample $x_n^{(\tau+1)} \sim p(x_n | x_1^{(\tau+1)}, x_2^{(\tau+1)}, \ldots, x_{n-1}^{(\tau+1)})$.
end for
return Last N sets of samples.

Gibbs Sampling Example

Bayesian Approach to Handle Missing Data:
Let $\mathbf{x}_{obs}$ denote the vector of observed data and $\mathbf{x}_{mis}$ the vector of missing data. We like to sample from $p(\boldsymbol{\theta}|\mathbf{x}_{obs})$.
Instead we sample from $p(\boldsymbol{\theta}, \mathbf{x}_{mis}|\mathbf{x}_{obs})$

Data Augmentation Algorithm

I-Step Generate $\mathbf{x}_{mis}^{(\tau+1)} \sim p(\mathbf{x}_{mis}|\boldsymbol{\theta}^{(\tau)}, \mathbf{x}_{obs})$

P-Step Generate $\boldsymbol{\theta}^{(\tau+1)} \sim p(\theta|\mathbf{x}_{mis}^{(\tau+1)}, \mathbf{x}_{obs})$

Question: What is the frequencist counterpart of this method?

Popularity of Gibbs Sampling

- Graphical models are defined using conditional distributions
- Easy to understand, easy to implement.
- Good trade-off between acceptance and mixing:
  - ⇒ Acceptance ratio is always 1.
- Open-source, black-box implementations!
  - ⇒ `BUGS` and `WinBUGS`
  - How:

$$p(x_i|\mathbf{x}_{-i}) = \frac{p(x_i, \mathbf{x}_{-i})}{\sum_{x_i'} p(x_i', \mathbf{x}_{-i})}$$

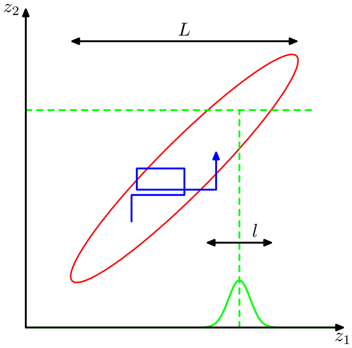


Figure Credit: Chris Bishop *PRML* 2006.

Acceleration of Gibbs Sampling

Task: Given $\tilde{p}(a, b, c)$ draw samples from a and c

Regular Gibbs:

- Draw a given b and c,
- Draw b given a and c,
- Draw c given a and b.

Blocked Gibbs:

- Draw (a, b) given c,
- Draw c given (a, b),

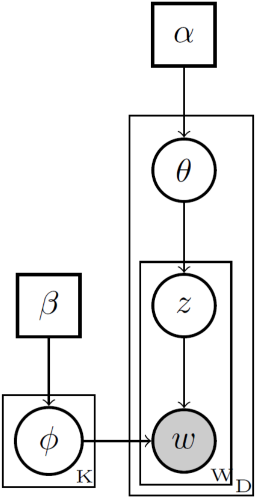
Collapsed Gibbs:

- Draw a given c,
- Draw c given a,

Message: Marginalize whenever you can!

Acceleration of Gibbs Sampling: The LDA example

- The model describes the joint probability of ϕ, θ and z.
- But we are interested only in inferring z (topic for each word).
- Marginalize the distribution for z.



Parallel Gibbs Sampling

The Synchronous Gibbs Sampler

```
for all x_i do
    In Parallel update x_i^(τ+1) ~ p(x_i | x_{-i}^(τ)).
end for
```

for all x_i **do**
    In Parallel update $x_i^{(\tau+1)} \sim p\left(x_i|\mathbf{x}_{-i}^{(\tau)}\right)$.
end for

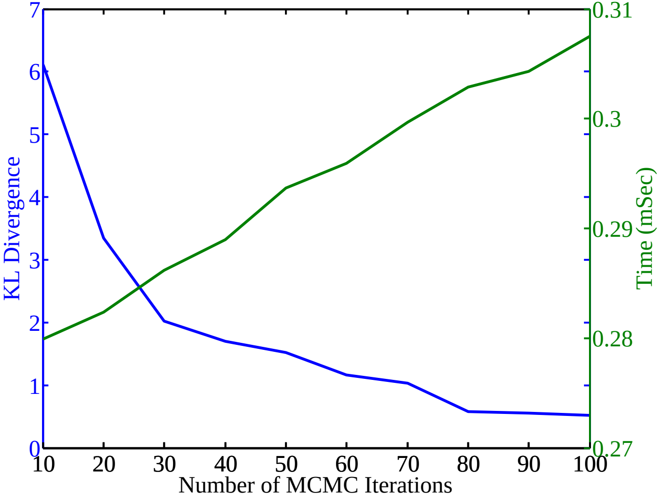
The Chromatic Sampler (2011)

Require: k-colored Graph

for all k colors in the graph **do**
    for all Variables $x_i \in \mathcal{G}_k$ **do**
        In Parallel update $x_i^{(\tau+1)} \sim p\left(x_i|N_{x_i}^{(\tau)}\right)$.
    end for
end for

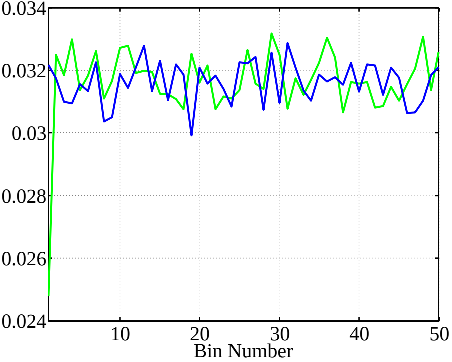
Convergence Diagnostics

Single Variable Distributions



Convergence Diagnostics I

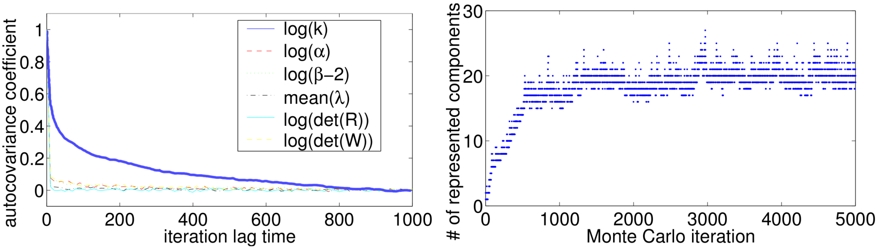
Multiple Variable Distributions



Burn-in Period?
Good initialization?

Convergence Diagnostics II

Analysis of Autocorrelation



Figures Credit: Rasmussen (2000).

For Diagnostics:

Standard Software Packages like `R-CODA`

In Practice:

Create a synthetic dataset and watch the accuracy of parameter estimation

Auxiliary Variable Methods

The General Approach to sample from $p(\mathbf{x})$:

- Specify auxiliary variables $\mathbf{u}$ and the conditional distribution $p(\mathbf{u}|\mathbf{x})$ to form the joint distribution $p(\mathbf{u}, \mathbf{x}) = p(\mathbf{u}|\mathbf{x})p(\mathbf{x})$.
- Sample from $(\mathbf{x}, \mathbf{u})$ using a MCMC algorithm.
- Computationally marginalize over $\mathbf{u}$ to obtain samples from $p(\mathbf{x})$.

When to use?

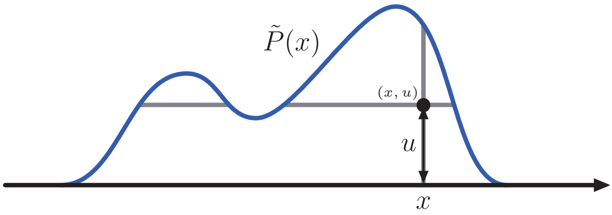
- The target distribution is multi-modal,
- Cancel the marginalization constant of the distribution.

How to choose the auxiliary variables?

- Very hard question. One of the common research topics.
- Depends on the problem. Look for physical meaning of the problem.

The Slice Sampling I

Task: Sample from $p(x) = \frac{1}{Z}\tilde{p}(x)$



Figures Credit: Murray (2009).

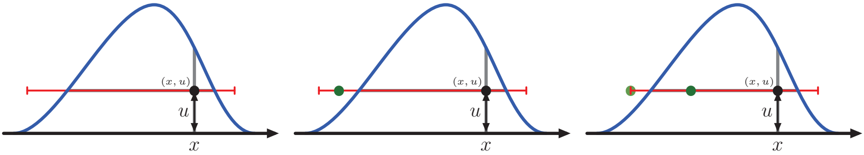
$$p(u|x) = \text{Uniform}\left[0, \tilde{p}(x)\right]$$

$$p(x|u) \propto \begin{cases} 1 & \text{if } \tilde{p}(x) \geq u \\ 0 & \text{Otherwise} \end{cases}$$

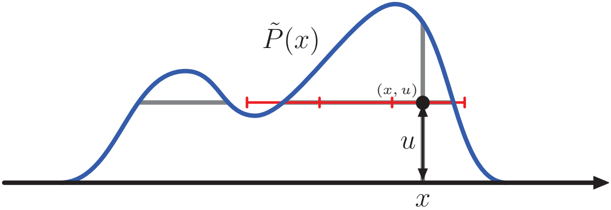
i.e. "Uniform on the slice".

The Slice Sampling II

Unimodal Case



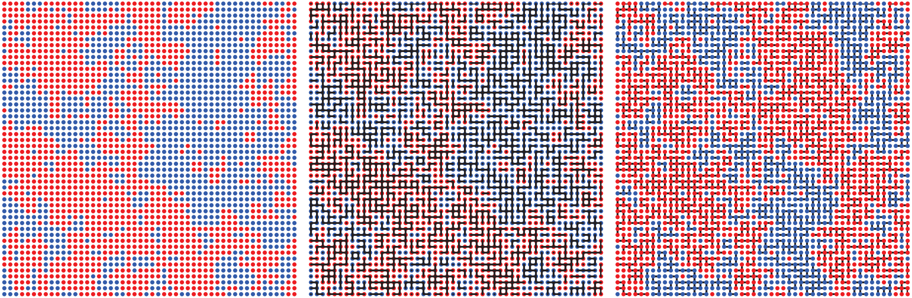
Multimodal Case



Figures Credit: Murray (2009).

The Swendsen-Wang Algorithm

$$p(\mathbf{x}) \propto \exp\left\{\sum_{i \neq j} \beta_{ij} \mathcal{I}[x_i = x_j]\right\},$$



Figures Credit: Murray (2009).

Importance Sampling based Algorithms

Importance Sampling I

DO NOT Throw samples away!
Weight them!

$$\int f(\mathbf{x})p(\mathbf{x})d\mathbf{x} = \int f(\mathbf{x})\frac{p(\mathbf{x})}{q(\mathbf{x})}q(\mathbf{x})d\mathbf{x}$$
$$= \int f(\mathbf{x})w(\mathbf{x})q(\mathbf{x})d\mathbf{x}$$
$$\approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}^{(i)})w(\mathbf{x}^{(i)}), \quad \mathbf{x}^{(i)} \sim q(\mathbf{x}).$$

where,

$$w(\mathbf{x}) = \frac{p(\mathbf{x})}{q(\mathbf{x})}.$$

Importance Sampling II

How to choose the proposal distribution $q(\mathbf{x})$?

- As similar as possible to $p(\mathbf{x})$.

Theorem The best proposal function is the following:

$$q^{\star}(\mathbf{x}) = \frac{|f(\mathbf{x})|p(\mathbf{x})}{\mathbb{E}_{p(\mathbf{x})}|f(\mathbf{x})|}.$$

Questions

- What is trivial in above formula?
- What are we optimizing?

Sequential Importance Sampling I

We want to sample from $p(x_{1:t})$.

- It is hard to sample from a multidimensional distribution.
- Sampling in real-time?

$\Rightarrow$ Choose a proposal function in the form of

$$q(x_{1:t}) = q(x_1) \prod_{k=2}^{t} q(x_k | x_{1:k-1}).$$

Use importance sampling. Nice iterative formula for the weights:

$$\begin{aligned} w(x_{1:t}) &= \frac{p(x_{1:t})}{q(x_{1:t})} \\ &= \frac{p(x_{1:t-1})}{q(x_{1:t-1})} \frac{p(x_{1:t})}{p(x_{1:t-1}) q(x_t | x_{1:t-1})} \\ w(x_{1:t}) &= w(x_{1:t-1}) \alpha_t \end{aligned}$$

Sequential Importance Sampling II

The Algorithm at step k

- Generate N samples $x_k^{(i)} \sim q(x_k|x_{1:k-1})$.
- Update the weights $w(x_{1:t}) = w(x_{1:t-1})\alpha_t$ for each sample $i = 1, \ldots, N$.

Problem Suppose we are at step k.

- The weight $w(k)^{(i)}$ for a particle is very small.
- The weights are updated in multiplicative way
  - $\Rightarrow$ weights will remain small.

Solution

- Throw the samples with tiny weights away?
- Replace them with the higher weighted samples $\Rightarrow$ Resampling.

Sequential Importance Reampling

The Algorithm at step k

- Generate N samples $x_k^{(i)} \sim q(x_k|x_{1:k-1})$.
- Update the weights $w(x_{1:t}) = \alpha_t$ for each sample $i = 1, \ldots, N$.
- Resample $x_k^{(i)}, i = 1, \ldots, N$ according to weights.

Resampling Algorithms

- Multinomial
- Systematic

Advantages

- Requires only one iteration to generate the samples.
- The generated samples are independent; no burn-in period or decoupling is required.
- Embarrassingly parallel (using GPUs)

Summary

- Sampling as an approximation
- Significance
- Rejection Sampling for unnormalized distributions
- Metropolis Hastings a very powerful and flexible MCMC Sampling algorithm
- Gibbs Sampler an easy to understand and easy to implement algorithm with many applications
- Additional improvement via auxiliary variables
- Practical Considerations.

Bibliography

General Overviews:

Video Lectures by Iain Murray on Markov Chain Monte Carlo (2009).

Andrieu, de Frietas and Jordan, (2003), *"An Introduction to MCMC for Machine Learning"*, Science.

Christopher Bishop, (2006), PRML Chapter 11

Brooks, S., Gelman, A., Jones, G., & Meng, X.-L. (2011). *"Handbook of Markov Chain Monte Carlo"*. Chapman and Hall/CRC.

Metropolis Hastings

Chib, S., & Greenberg, E. (1995). *"Understanding the Metropolis-Hastings Algorithm"*. The American Statistician.

Gibbs Sampling:

Casella, G., & George, E. I. (1992). *"Explaining the Gibbs Sampler"*. The American Statistician.

Convergence Diagnostics

Geyer, C. J. (1992). *"Practical Markov Chain Monte Carlo"*. Statistical Science.

Auxiliary Variable Methods

Gray, A. J. (1994). *"Simulating posterior Gibbs distributions: a comparison of the Swendsen-Wang and Gibbs sampler methods"*. Statistics and Computing.

Sequential Monte Carlo

Doucet, A., & Johansen, A. M. (2008). *"A Tutorial on Particle Filtering and Smoothing: Fifteen years Later."*

If I had time ...

- Importance Sampling
- Exact Sampling, decoupling from the past
- Multiple-Try Metropolis
- Hybrid Sampling methods, ex. Hamiltonian Monte Carlo (HMC)
- Reversible-jump MCMC
- More examples of auxiliary variable methods: ex. Annealing, Tempering, etc.

Thank you!