Monte Carlo Sampling and MCMC (Part I)

Advanced Topics in Deep Learning

Vahid Tarokh

ECE 590-02

Spring 2022

Motivation

- Key to generative models we have discussed (and those that will be further covered in this course) is sampling from a distribution that estimates that of data points.
- Up to know, we intentionally designed these distributions such that sampling from them is easy, e.g. GANs, Normalizing Flows, RBMs, VAEs, etc.
- We also made approximations to make the training made simpler.
- In order to present more advanced generative methods (and also for other reasons we will discuss later), we need to briefly discuss **Sampling** from given distributions.
- In the **first part** of Lectures on Monte-Carlo sampling, <u>we do not consider deep versions</u>. **We will revisit these later.**

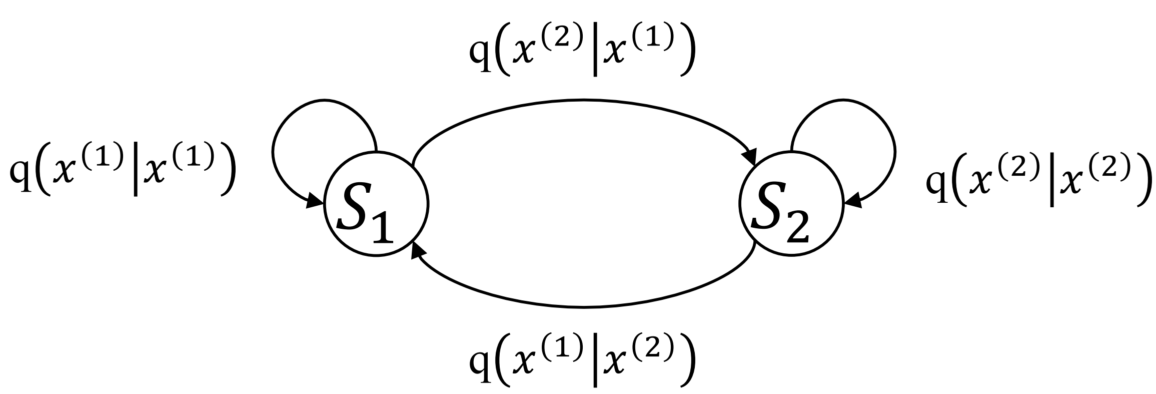
Outline

- Background and Related Work
  - Quick review of Markov Chains (prepared by Kevin Choy)
  - Monte Carlo Principal
  - Importance Sampling
  - Sequential Monte Carlo (SMC)
  - **MCMC**
    - Metropolis-Hastings Algorithm
    - Random-Walk Metropolis
    - Gibbs Sampling.
    - Hamiltonian Monte Carlo (HMC)
    - Metropolis Adjusted Langevin Algorithm (MALA) also known as Langevin Monte Carlo (LMC)

Quick Review of Markov Chains

Background on Markov Chains

- A stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.
- Markov Property:
  - $q(\boldsymbol{x}^{(t)}|\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t-2)}, \dots, \boldsymbol{x}^{(1)}) = q(\boldsymbol{x}^{(t)}|\boldsymbol{x}^{(t-1)})$
  - $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(t)})$ represent sequentially drawn samples at discrete times t.



Background on Markov Chains

Multiple types of Markov Chains used for different applications – all have Markov property

- **Discrete Time Discrete Space**
  - Finite time steps indexed by integers with finite number of states (we will focus on this for the initial understanding)
  - E.g., Turn based game with finite positions
- Discrete Time Continuous Space
  - Finite time steps with infinite states
  - E.g., Particle location at discrete observations
- Continuous Time Discrete Space
  - Infinite time steps with finite number of spaces
  - E.g., Particle oscillating continuously between two states
- Continuous Time Continuous Space
  - Infinite time steps with infinite spaces
  - E.g., Particle in physical space at physical time

Background on Markov Chains

- Markov chain: $\mathrm{q}\left(\boldsymbol{x}^{(t)} \middle| \boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t-2)}, \dots, \boldsymbol{x}^{(1)}\right) = q\left(\boldsymbol{x}^{(t)} \middle| \boldsymbol{x}^{(t-1)}\right)$.
  - $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(t)})$ represent sequentially drawn samples at discrete times t.
  - $p(\cdot|\cdot)$: transition distribution.
- Markov chains are characterized by:
  - A state space S, finite or countable set of values that the random variables may take. $S = \{1, 2, 3, \dots\}$
  - Transition matrix q describing the probabilities of particular transitions between states
  - Initial distribution π_0, denoting the distribution of the Markov chain at time 0.
    - $\pi_0(i)$ denotes the probability that the Markov chain starts out in state i for each state $i \in S$.

Some Properties of Markov Chains

- Aperiodicity
  - A state has period k if any return to this state must occur in multiples of k time steps
  - $k = gcd\{n: q(x^{(n)} = i | x^{(0)} = i) > 0\}$
  - If $k = 1$, the state is said to be *aperiodic*
  - If all states are aperiodic, the chain is considered aperiodic
- Irreducibility
  - Two states communicate with each other if both are accessible from one another
  - A Markov chain is irreducible all pairs of states communicate

Some Properties of Markov Chains

- Ergodicity
  - Ergodic state means the state is visited more than once with probability 1
  - If all states are ergodic the chain is considered ergodic
  - Any state can be reached from any other state in less than finite number of steps

Some Properties of Markov Chains

- Transience
  - A state is transient if there's a non-zero probability that the state will not be revisited
- Recurrence
  - A state is recurrent if, starting from this state at time 0, the chain will eventually return to this state

Modeling Long-Term Behavior

- Transitions between different states of a Markov chain describe *short-time* behavior of the chain.
- We want to model the long-term behavior of Markov chains
- Let $\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(t)}\right)$ be a finite-state, irreducible and aperiodic Markov chain, then the limiting distribution exists: $\lim_{t \to \infty} p_t = \pi$
- The limiting distribution of the chain is the *stationary distribution.*

Stationary Distributions

- A distribution $\pi = \left(\pi(i)\right)_{\mathrm{i} \in S}$ on state space S of a Markov chain is a stationary distribution if:
  - $Pr\left(\boldsymbol{x}^{(2)} = i\right) = \pi(i)$ for all $i \in S$, whenever $Pr\left(\boldsymbol{x}^{(1)} = i\right) = \pi(i)$ for all $i \in S$
  - i.e., the distribution of $\boldsymbol{x}^{(2)}$ is equal to the distribution of $\boldsymbol{x}^{(1)}$ when the distribution of $\boldsymbol{x}^{(1)}$ is π.

Stationary Distributions

- A nonnegative vector $\pi = \left(\pi(i)\right)_{i\in S}$ with $\sum_{i\in S}\pi(i) = 1$ is a stationary distribution if and only if $\pi = \pi q$.
  - Here π is interpreted as a row vector.
- In that case the Markov chain with initial distribution π and transition matrix q is stationary and the distribution of $\boldsymbol{x}^{(t)}$ is π.

Stationary Distributions

- Basic Limit Theorem
  - Let $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(t)})$ be an irreducible, aperiodic Markov chain having a stationary distribution $\pi(\cdot)$. Let $\boldsymbol{x}^{(0)}$ have the distribution π_0, an arbitrary initial distribution. Then $\lim_{t \to \infty} \pi_t(i) = \pi(i)$ for all states i.
- Markov Chain Monte Carlo (MCMC)
  - Main Idea: create a Markov chain with stationary distribution equal to target distribution.

Additional Resources

- Prof. Joe Chang's Stochastic Processes notes
  - http://www.stat.yale.edu/~pollard/Courses/251.spring2013/Handouts/Chang-MarkovChains.pdf
- Material on Stationary and Limiting Distributions by Prof. Gordan Žitković
  - https://web.ma.utexas.edu/users/gordanz/notes/stationary_distributions_color.pdf

Discussion of Monte Carlo Methods

The Monte Carlo principle

- p(x): a target density
- Monte Carlo techniques draws a set of (iid) samples $\{x^1,\ldots,x^N\}$ from p in order to approximate p with the empirical distribution

$$p(x) \approx \frac{1}{N}\sum_{i=1}^{N}\delta(x = x^{(i)})$$

- Using these samples we can approximate expectations with tractable empirical sums that converge to the true expectation, e.g.

$$\int f(x)p(x)dx \approx \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})$$

Importance Sampling

- *p(x)* is known, and we want to compute

$$\int f(x)p(x)dx$$

- We introduce another an auxiliary (**proposal**) density that its support is a superset of the support of p. Then:

$$\int f(x) \underbrace{p(x)/q(x)}_{w(x)\text{ 'importance weight'}} * q(x)dx \approx \sum_{i=1}^{N} f(x^{(i)})w(x^{(i)})$$

- Idea: Sample from q instead of p and
  - Weight the samples according to their importance as above
- Key issue is a ‘good’ choice of q.
  - Sampling from q must be easy and calculations must not be costly

Sequential Monte Carlo (SMC)

- SMC is an online algorithm whose goal is to estimate the distribution $p(x_{0:t} \mid y_{1:t})$, where
  - y_t is observation at each time t
  - $x_{0:t}$ are hidden parameters/states that must be estimated
  - We have a model:
    - Initial distribution: $p\left(x_0\right)$
    - Dynamic model: $p\left(x_t | x_{0:t-1}, y_{1:t-1}\right)$ for $t \geq 1$
    - Measurement model: $p\left(y_t | x_{0:t}, y_{1:t-1}\right)$ for $t \geq 1$

Sequential Monte Carlo (SMC)

- We define a ***proposal*** distribution:

$$q(\widetilde{x}_{0:t}|y_{1:t}) = p(x_{0:t-1}|y_{1:t-1})q(\widetilde{x}_t|x_{0:t-1}, y_{1:t})$$

- Then the importance weights are:

$$\begin{aligned} w_t &= \frac{p(\widetilde{x}_{0:t}|y_{1:t})}{q(\widetilde{x}_{0:t}|y_{1:t})} = \frac{p(x_{0:t-1}|y_{1:t})}{p(x_{0:t-1}|y_{1:t-1})} \frac{p(\widetilde{x}_t|x_{0:t-1}, y_{1:t})}{q(\widetilde{x}_t|x_{0:t-1}, y_{1:t})} \\ &\propto \frac{p\left(y_t|\widetilde{x}_t\right) p\left(\widetilde{x}_t|x_{0:t-1}, y_{1:t-1}\right)}{q_t\left(\widetilde{x}_t|x_{0:t-1}, y_{1:t}\right)}. \end{aligned}$$

- Please note that by simplifying choice for proposal distribution: $q(\widetilde{x}_t|x_{0:t-1}, y_{1:t}) = p\left(\widetilde{x}_t|x_{0:t-1}, y_{1:t-1}\right)$ we arrive at $w_t \propto p\left(y_t|\widetilde{x}_t\right)$
  - This is intuitively appealing.

Sequential Monte Carlo (SMC)

Sequential importance sampling step

– For $i = 1, \ldots, N$, sample from the transition priors

$$\widetilde{x}_t^{(i)} \sim q_t\left(\widetilde{x}_t | x_{0:t-1}^{(i)}, y_{1:t}\right)$$

and set

$$\widetilde{x}_{0:t}^{(i)} \triangleq \left(\widetilde{x}_t^{(i)}, x_{0:t-1}^{(i)}\right)$$

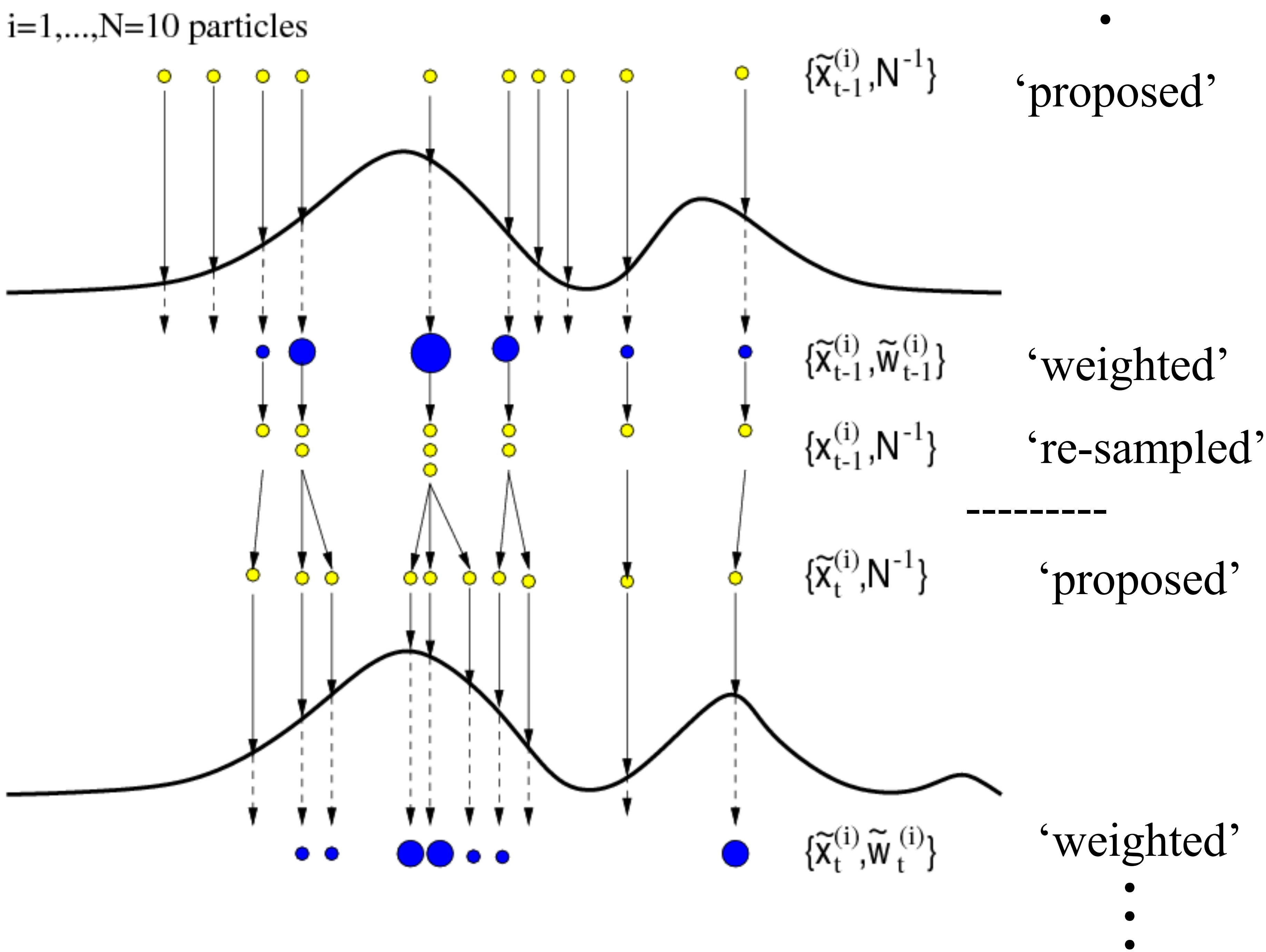
– For $i = 1, \ldots, N$, evaluate and normalize the importance weights

$$w_t^{(i)} \propto \frac{p\left(y_t|\widetilde{x}_t^{(i)}\right) p\left(\widetilde{x}_t^{(i)}|x_{0:t-1}^{(i)}, y_{1:t-1}\right)}{q_t\left(\widetilde{x}_t^{(i)}|x_{0:t-1}^{(i)}, y_{1:t}\right)}.$$

Selection step

– Multiply/Discard particles $\left\{\widetilde{x}_{0:t}^{(i)}\right\}_{i=1}^{N}$ with high/low importance weights $w_t^{(i)}$ to obtain N particles $\left\{x_{0:t}^{(i)}\right\}_{i=1}^{N}$.

Graphical Example



Markov Chain Monte Carlo

- Main Idea: create a Markov chain with stationary distribution equal to target distribution.
  - Simulate the chain long enough that samples eventually come from the stationary distribution.

Markov Chain Theory (Very Short Review)

- Markov chain: $q\left(\boldsymbol{x}^{(t)} \middle| \boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t-2)}, \dots, \boldsymbol{x}^{(1)}\right) = q\left(\boldsymbol{x}^{(t)} \middle| \boldsymbol{x}^{(t-1)}\right)$.
  - $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(t)})$ represent sequentially drawn samples at discrete times t.
  - $p(\cdot|\cdot)$: transition distribution.
- π is the stationary or target distribution of a Markov chain with transition probabilities p if $\pi = \pi q$
- Need to prove:
  1. Law of large numbers for dependent samples from Markov chain.
  2. Stationary distribution of Markov chain exists.

Ergodic Theorem

- **Theorem 1** (Ergodic Theorem). *If a Markov chain is ergodic and* $E_\pi[f(\boldsymbol{x})] < \infty$ *for the unique target distribution* π, *then*

$$\frac{1}{N}\sum_{i=1}^{N} f\left(\boldsymbol{x}^{(i)}\right) \xrightarrow{a.s.} E_\pi[f(\boldsymbol{x})],$$

as $N \to \infty$.

- Conditions for a Markov chain to be ergodic:
  - Aperiodicity, recurrence and irreducibility.

Detailed Balance

- **Theorem 2** (Detailed Balance). *Suppose a Markov chain with transition distribution* $q(\cdot|\cdot)$ *satisfies the detailed balance condition with probability density function* π:

$$\mathrm{q}(y|x)\pi(x) = q(x|y)\pi(y) \forall x, y.$$

Then π is the stationary distribution of the Markov chain and the chain is reversible.

Markov Chain Theory Summary

- If we can can choose $q(\cdot|\cdot)$ such that the target distribution is the stationary distribution of an ergodic Markov chain:
  - **Sampling from the Markov chain is asymptotically the same as sampling from the target distribution.**
  - No matter initial starting point, will eventually get samples from the target distribution.
  - Can use collection of samples from the Markov chain to summarize the target distribution.

- In practice, common MCMC algorithms designed so the Markov chain is ergodic and satisfies detailed balance.

Metropolis-Hastings Algorithm

- Defines the acceptance probability such that the Markov chain satisfies detailed balance when combined with an arbitrary proposal distribution, $q(\cdot | \cdot)$.
  - $q(\cdot | \cdot)$ relates to transition probability discussed above.

- Don't need to know the normalizing constant of the posterior.
- Can be used with discrete and continuous parameters, in general.
  - Wide range of choices for the proposal distribution, leading to different convergence rates.

Metropolis-Hastings Algorithm

1. Initialise $x^{(0)}$.
2. For $i = 0$ to $N - 1$
   - Sample $u \sim \mathcal{U}_{[0,1]}$.
   - Sample $x^{\star} \sim q(x^{\star}|x^{(i)})$.
   - If $u < \mathcal{A}(x^{(i)}, x^{\star}) = \min\left\{1, \frac{p(x^{\star})q(x^{(i)}|x^{\star})}{p(x^{(i)})q(x^{\star}|x^{(i)})}\right\}$

     $$x^{(i+1)} = x^{\star}$$

     else

     $$x^{(i+1)} = x^{(i)}$$

The Metropolis algorithm assumes a symmetric random walk proposal $q(x^{\star}|x^{(i)}) = q(x^{(i)}|x^{\star})$ and, hence, the acceptance ratio simplifies to

$$\mathcal{A}(x^{(i)}, x^{\star}) = \min\left\{1, \frac{p(x^{\star})}{p(x^{(i)})}\right\}.$$

MCMC Challenges and Extensions

- Basic random-walk Metropolis-Hastings can be very slow to converge.
  - Especially for highly correlated parameters in the posterior.
  - Reparameterization or auxiliary variables can sometimes help.
  - Multi-modal posterior distributions.
    - Simulated tempering methods.
- Other extensions:
  - Slice-sampling, reversible-jump sampling, sequential Monte Carlo, genetic algorithms.
  - Approaches to rapidly explore the posterior by suppressing random-walk behavior.

Gibbs Sampling

- Component-wise proposal q:

$$q(x^\star|x^{(i)}) = \begin{cases} p(x_j^\star|x_{-j}^{(i)}) & \text{If } x_{-j}^\star = x_{-j}^{(i)} \\ 0 & \text{Otherwise.} \end{cases}$$

Where the notation means:

$$p(x_j|x_{-j}) = p(x_j|x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_n)$$

- In this case, the acceptance probability is $\mathcal{A}(x^{(i)},x^\star)=1$

The Gibbs Sampling Algorithm

1. Initialise $x_{0,1:n}$.
2. For $i = 0$ to $N - 1$
   - Sample $x_1^{(i+1)} \sim p(x_1 | x_2^{(i)}, x_3^{(i)}, \ldots, x_n^{(i)})$.
   - Sample $x_2^{(i+1)} \sim p(x_2 | x_1^{(i+1)}, x_3^{(i)}, \ldots, x_n^{(i)})$.

$$\vdots$$

   - Sample $x_j^{(i+1)} \sim p(x_j | x_1^{(i+1)}, \ldots, x_{j-1}^{(i+1)}, x_{j+1}^{(i)}, \ldots, x_n^{(i)})$.

$$\vdots$$

   - Sample $x_n^{(i+1)} \sim p(x_n | x_1^{(i+1)}, x_2^{(i+1)}, \ldots x_{n-1}^{(i+1)})$.

Hamiltonian Monte Carlo (HMC)

- As before, suppose the target distribution to sample is $p(x)$.
- The Hamiltonian (which comes from Newtonian Mechanics) is defined by

$$H(x,p) = U(x) + \frac{1}{2}\boldsymbol{p}^T M^{-1}\boldsymbol{p}$$

where M is the mass matrix which is symmetric and positive, $\boldsymbol{p}$ is the momentum, x is the position and $U(x)$ is the potential energy.

Hamiltonian Monte Carlo

- If $p(x)$ is the target distribution, then we let $U(x) = -\ln(p(x))$. Thus
$$p(x) = \exp(\text{-}\,U(x)\,).$$
  - This is related to Boltzmann distribution.
- The algorithm fixes an integer L > 0 referred to as number of leap-frog steps and a step size Δt.
- Suppose the chain is at $\boldsymbol{X_n} = x_n$. It (sets the initial state of the leap-frog to $x_n(0) = x_n$; It also samples a random momentum $p_n(0)$ according to Gaussian distribution $N(0, M)$.

HMC Algorithm

- Next we recall the Hamilton Equations

$$\frac{dx}{dt} = \frac{\partial H}{\partial \boldsymbol{p}},$$

and

$$\frac{d\boldsymbol{p}}{dt} = - \frac{\partial H}{\partial x}.$$

- We will use the above and finite difference approximations to derivatives in every leap-from step, combined with $U(x) = -\ln(p(x))$. Discretization gives the following update rule.
- The particle under Hamiltonian dynamics for L Δt seconds corresponding to L leap-frog states of Δt seconds each.

HMC Algorithm

- The finite difference to Hamiltonian equations leads to the following updates:

$$\mathbf{p}_n\left(t+\frac{\Delta t}{2}\right) = \mathbf{p}_n(t) - \frac{\Delta t}{2}\nabla U(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_n(t)}$$
$$\mathbf{x}_n(t+\Delta t) = \mathbf{x}_n(t) + \Delta t M^{-1}\mathbf{p}_n\left(t+\frac{\Delta t}{2}\right)$$
$$\mathbf{p}_n(t+\Delta t) = \mathbf{p}_n\left(t+\frac{\Delta t}{2}\right) - \frac{\Delta t}{2}\nabla U(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_n(t+\Delta t)}$$

- These finite difference equations when applied to $x_n(0)$ and $\boldsymbol{p}_n(0)$ give $x_n(\mathrm{L}\,\Delta \mathrm{t})$ and $\boldsymbol{p}_n(\mathrm{L}\,\Delta \mathrm{t})$.

- The Hamiltonian Monte-Carlo (HMC) now uses the Metropolis-Hasting update technique in order to guarantee the convergence of stationary distribution to $p(x)$.

HMC Algorithm

- The transition from $\boldsymbol{X_n} = x_n$ to $\boldsymbol{X_{n+1}}$ is given by the Metropolis-Hasting update:

$$\mathbf{X}_{n+1}|\mathbf{X}_n = \mathbf{x}_n = \begin{cases} \mathbf{x}_n(L\Delta t) & \text{with probability } \alpha\left(\mathbf{x}_n(0), \mathbf{x}_n(L\Delta t)\right) \\ \mathbf{x}_n(0) & \text{otherwise} \end{cases}$$

where the acceptance probability is given by:

$$\alpha\left(\mathbf{x}_n(0), \mathbf{x}_n(L\Delta t)\right) = \min\left(1, \frac{\exp[-H(\mathbf{x}_n(L\Delta t), \mathbf{p}_n(L\Delta t))]}{\exp[-H(\mathbf{x}_n(0), \mathbf{p}_n(0))]}\right)$$

- This process is repeated for $\boldsymbol{X_{n+1}}, \boldsymbol{X_{n+2}}, \cdots$
- Under mild assumption the limiting distribution can be proved to be $p(x)$.

Metropolis Adjusted Langevin Algorithm (MALA)

- MALA is of interest due to its simplicity.
- Updates for iteration $t+1$:

$$\tilde{x}^{(n+1)} = x^{(n)} + \frac{\boldsymbol{\epsilon}}{2}\nabla \ln(p\big(x^{(n)}\big)) + N(\mathbf{0}, \epsilon \boldsymbol{I})$$

- This proposal is accepted or rejected according to probability $\min\left(1, \frac{p\left(\tilde{x}^{(n+1)}\right) q\left(x^{(n)} | \tilde{x}^{(n+1)}\right)}{p\left(x^{(n)}\right) q\left(\tilde{x}^{(n+1)} | x^{(n)}\right)}\right)$, where $q(y|x) \propto \exp(-\left| \frac{y - x - \frac{\epsilon}{2}\nabla \ln\left(p\left(x^{(n)}\right)\right)}{2\epsilon}\right|^2)$.
- Under mild assumption the limiting distribution can be proved to be $p(x)$.