Markov-Chain Monte-Carlo

Advanced Seminar "Machine Learning"

Sascha Meusel

04.02.2015

Winter Semester 2014/2015

Motivation

What is Markov-Chain Monte-Carlo, and what use has it?

- Problems can be difficult to solve analytically, or don't even have any analytical solution
- MCMC is a class of algorithms based on Monte Carlo sampling, tackling such problems

- for Monte Carlo: needed distributions can be difficult to implement (e.g. non-Gaussian / non-Uniform)
- but Markov chains can provide also complexer distributions
- Markov chains are a kind of state machines with transitions to other states having a certain probability
- Starting with an initial state, calculate the probability which each state will have after N transitions
  $\rightarrow$ distribution over states

Motivation

Example: calculate volume of d-dimensional convex body

- Solution with MCMC: Formulate distribution over $x \in \mathbb{R}^d$ with

$$p(x) = \begin{cases} 1 & \text{if } x \text{ inside body} \\ 0 & \text{else} \end{cases}$$

- Draw N samples x_i of a d-dimensional bounding box BB in $\mathbb{R}^d$ with the convex body completely inside it
- The volume of the bounding box is known $(side_1 * side_2 * ... * side_d)$
- $\frac{|\text{samples inside box}|}{N} * volume(BB) \approx volume(body)$

In this simple example no Markov chain usage is visible, but there exists more sophisticated MCMC methods using Markov chains to solve this problem.

Contents

Introduction to Monte-Carlo

Task: expectation value needed:

$$\mathbb{E}_{p(x)}[f(x)] = \int f(x)p(x)\,dx$$

Problem: no or only an expensive analytical solution
Solution: Sample over $p(x)$:

$$\mathbb{E}_{p(x)}[f(x)] \approx \hat{f} = \frac{1}{S}\sum_{s=1}^{S} f(x^{(s)}), \quad x^{(x)} \sim p(x)$$

Introduction to Monte-Carlo

Properties:
Unbiased estimator $\hat{f}$:

$$\mathbb{E}_{p(\{x^{(s)}\})}\left[\hat{f}\right] = \sum_{s=1}^{S} \mathbb{E}_{p(x)}[f(x)] = \mathbb{E}_{p(\{x^{(s)}\})}\left[f(x)\right]$$

Variance shrinks $\propto \frac{1}{S}$:

$$var_{p(\{x^{(s)}\})}\left[\hat{f}\right] = \frac{1}{S^2} \sum_{s=1}^{S} var_{p(x)}[f(x)] = \frac{1}{S} var_{p(\{x^{(s)}\})}\left[f(x)\right]$$

Introduction to Markov-Chains

Markov chain on finite state space:

- stochastic process $x^{(i)} \in \mathcal{X} = \{x_1, ..., x_S\}$ (sequence of random variables)
- $p(x^{(i)}|x^{(i-1)}, ..., x^{(1)}) = T(x^{(i)}|x^{(i-1)})$
  $\rightarrow$ T depends only on current state $i-1$

homogeneous Markov chain:

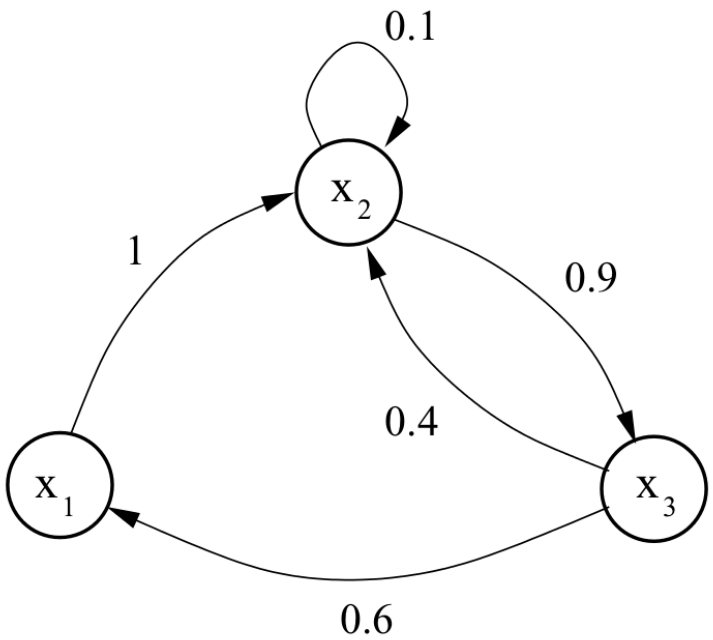
- T is invariant $\forall i$, with $\sum_{x^{(i)} \in \mathcal{X}} T(x^{(i)}|x^{(i-1)}) = 1 \quad \forall i$
- $\rightarrow$ fixed T matrix, with
  $p(x^{(i)}|x^{(i-1)}, ..., x^{(1)}) = Tp(x^{(i-1)}|x^{(i-2)}, ..., x^{(1)})$
- given **irreducibility** and **aperiodity**, chain converges to invariant distribution $p(x)$ after several steps:
  $p_N(x) = T^N p_0(x)$

Introduction to Markov-Chains

$$T = \begin{bmatrix} 0 & 0 & 0.6 \\ 1 & 0.1 & 0.4 \\ 0 & 0.9 & 0 \end{bmatrix}, \text{ initial distribution: } p_0(x) = \begin{pmatrix} 0.5 \\ 0.2 \\ 0.3 \end{pmatrix}$$

$T_{i,j}$: the probability to go to state i given state j

$$p_N(x) = T^N p_0(x), \quad \text{gives} \quad p_N(x) = \begin{pmatrix} 0.2 & 0.4 & 0.4 \end{pmatrix}^T$$



Introduction to Markov-Chains

Markov chain on continuous state space:

$$\int p(x^{(i)})K(x^{(i+1)}|x^{(i)})\,dx^{(i)} = p(x^{(i+1)})$$

- instead T an integral kernel K: the conditional density of $x^{(i+1)}$ given $x^{(i)}$
- is a mathematical description for a Markov chain algorithm

Metropolis-Hastings

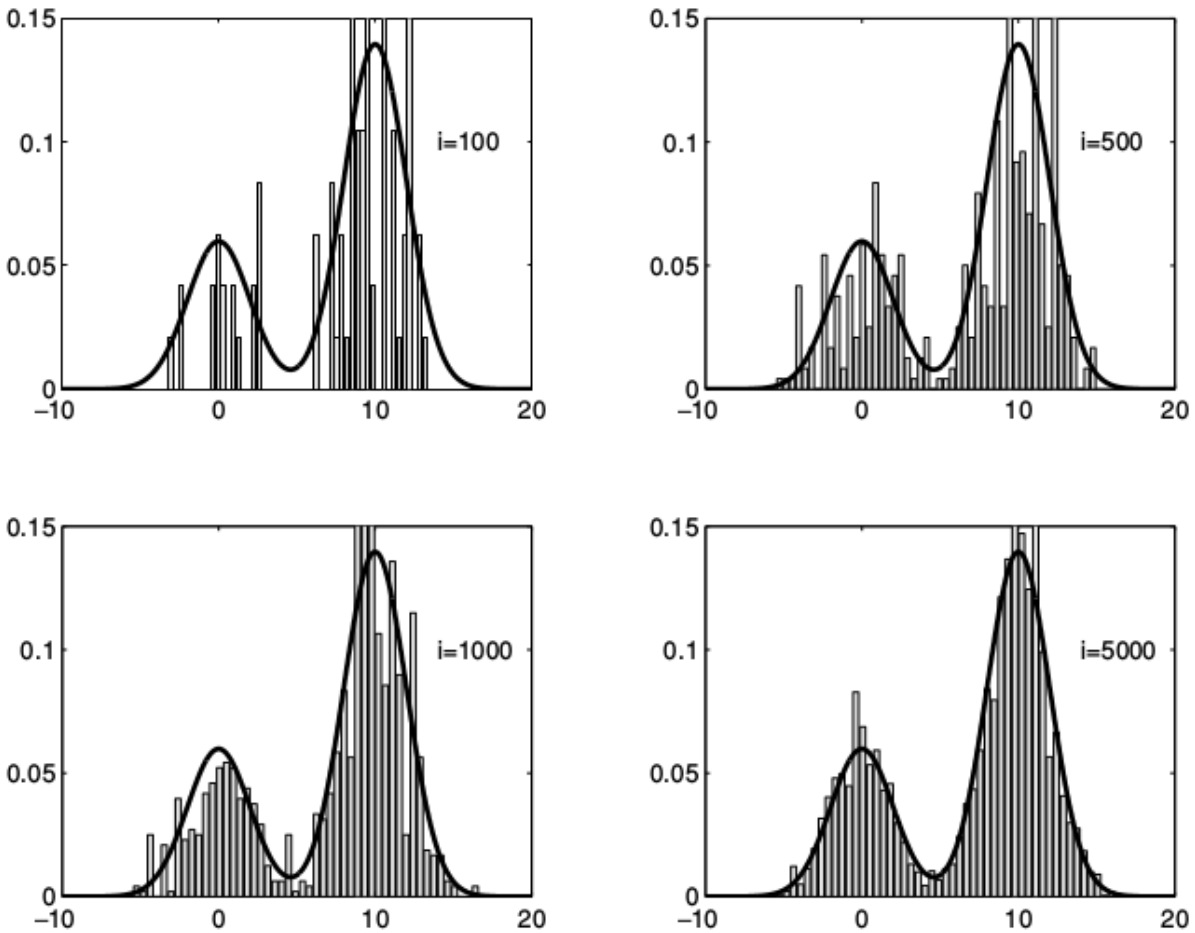
proposal distribution $q(x^*|x)$, with x^* being a sampling candidate and x being the current value

target distribution $p(x)$

acceptance probability $\mathcal{A}(x^{(i)}, x^*) = min\left(1, \frac{p(x^*)q(x^{(i)}|x^*)}{p(x^{(i)})q(x^*|x^{(i)})}\right)$

- initialize $x^{(0)}$
- for $i = 0$ to $N - 1$:
  - sample $u \sim \mathcal{U}_{[0,1]}$ $\quad //\mathcal{U}$ is uniform distribution
  - sample $x^* \sim q(x^*|x^{(i)})$
  - if $u < \mathcal{A}(x^{(i)}, x^*)$:
    - $x^{(i+1)} = x^*$
  - else:
    - $x^{(i+1)} = x^{(i)}$

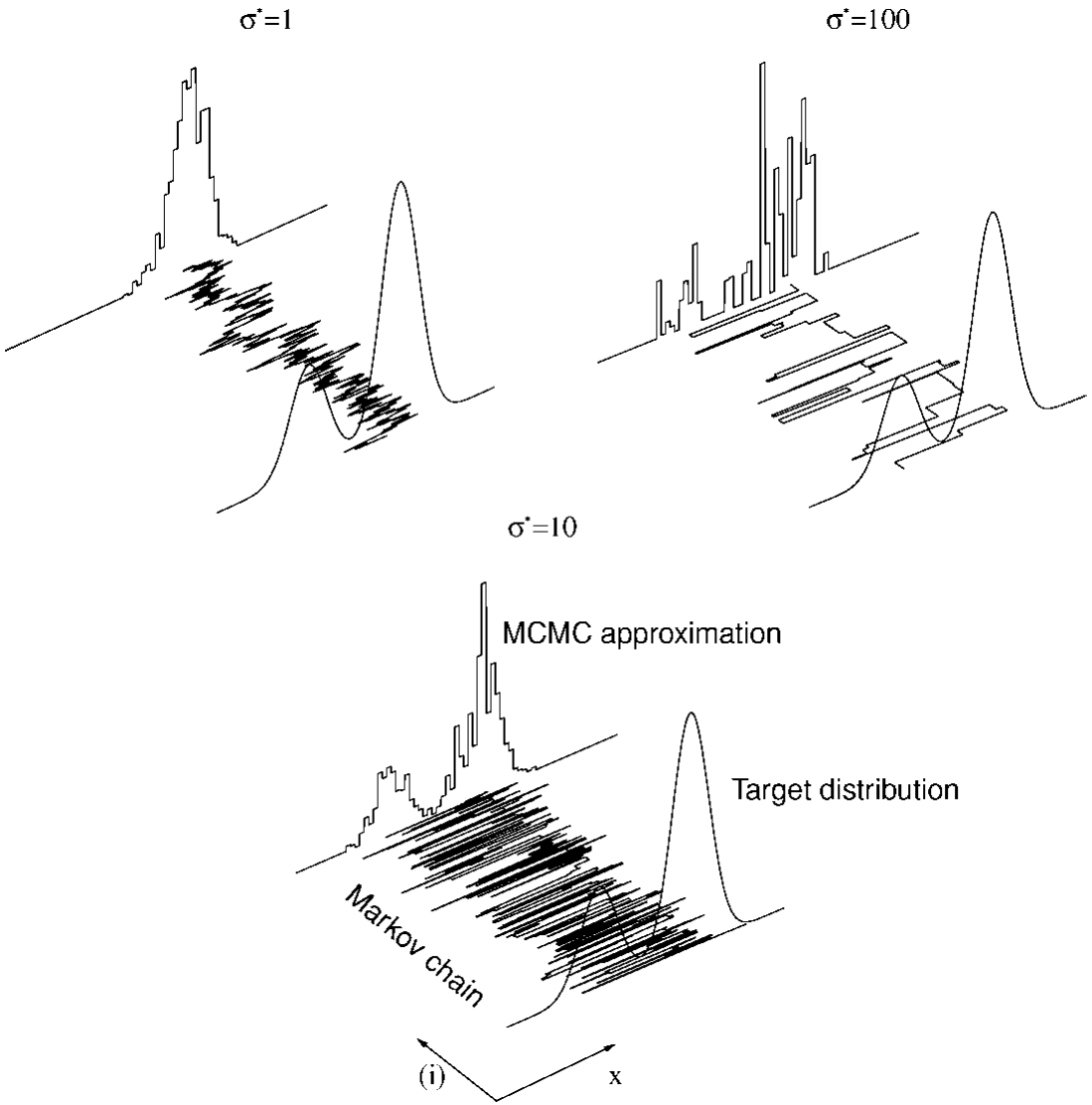
Metropolis-Hastings



proposal distribution $q(x^*|x^{(i)}) = \mathcal{N}(x^{(i)}, 100)$

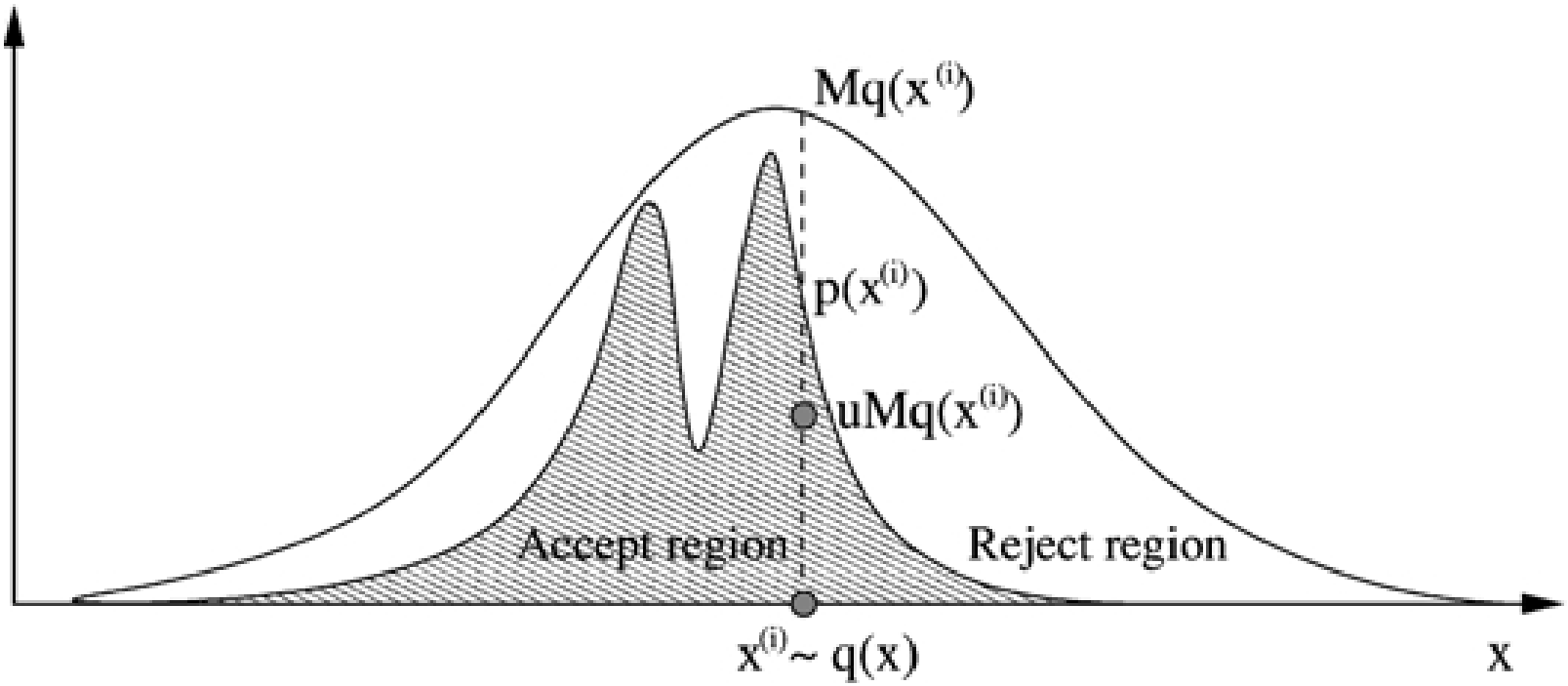
bimodal target distribution $p(x) \propto 0.3e^{-0.2x^2} + 0.7e^{-0.2(x-10)^2}$

Metropolis-Hastings



Rejection Sampling

Given: complex distribution $p(x)$
Choose a distribution $q(x)$ which we can sample (e.g. Gaussian)
Find factor M, so that $p(x) \leq Mq(x)$, with $M < \infty$



Rejection Sampling

Sampling algorithm:

- $i := 1$
- while $i \leq N$:
  - sample $x^{(i)} \sim q(x)$
  - sample $u \sim \mathcal{U}_{(0, Mq(x^{(i)}))}$
  - if $u < p(x^{(i)})$:
    - accept $x^{(i)}$ as sample
    - $i++$
  - else: reject sample

To avoid too many rejections, $Mq(x)$ should be chosen so that it bounds p(x) as strong as possible.

Importance Sampling

$$\int f(x)p(x)\,dx = \int f(x)\frac{p(x)}{q(x)}q(x)\,dx$$

$$\approx \frac{1}{S}\sum_{s=1}^{S} f(x^{(s)})\frac{p(x^{(s)})}{q(x^{(s)})}, \text{ with } x^{(s)} \sim q(x)$$

$\frac{p(x^{(s)})}{q(x^{(s)})}$ is the importance weight $w^{(s)}$

So we can simply sample over $q(x)$ and multiply each sample with its weight $w^{(s)} \rightarrow$ no rejections

Gibbs sampling

Let x be n-dimensional.
Also let be given, that we can calculate

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

with a proposal distribution $q(x^*|x^{(i)}) = \begin{cases} p(x_j^*|x_{-j}^{(i)}) & \text{if } x_{-j}^* = x_{-j}^{(i)} \\ 0 & \text{else} \end{cases}$

and $\mathcal{A}(x^{(i)}, x^*) = min\left(1, \frac{p(x^*)q(x^{(i)}|x^*)}{p(x^{(i)})q(x^*|x^{(i)})}\right) = min\left(1, \frac{p(x_{-j}^*)}{p(x_{-j}^{(i)})}\right) = 1$

- initialize $x_{1:n}^{(0)}$
- for $i = 0$ to $N - 1$:
  - for $j = 0$ to n:
    - $x_j^{(i+1)} \sim p(x_j|x_1^{(i)}, ..., x_{j-1}^{(i)}, x_{j+1}^{(i)}, ..., x_n^{(i)})$

Hybrid Monte Carlo

Also known as Hamilton Monte Carlo.
Basic idea: using gradient of target distribution

- Simulate walk through target distribution as a sphere without friction on a potential field surface.
- Therefore auxiliary variables $u \in \mathbb{R}^{n_x}$ needed to store momentum of sphere.
- Sphere will be more often in areas with lower potential, so those areas represents regions in the target distribution with higher density.
- parameters: stepsize ρ and number of steps per iteration L

Hybrid Monte Carlo

- initialize $x^{(0)}$
- for $i = 0$ to $N - 1$:
  - sample $v \sim \mathcal{U}_{[0,1]}$ and $u^* \sim \mathcal{N}(0, I_{n_x})$
  - define $x_0 = x^{(i)}$ and $u_0 = u^* + \rho\Delta(x_0)/2$
  - for $l = 1$ to L:
    - $x_l = x_{l-1} + \rho u_{l-1}$
    - $u_l = u_{l-1} + \rho_l \Delta(x_l)$ with $\rho_l = \begin{cases} \rho & \text{if } l < L \\ \rho/2 & \text{if } l = L \end{cases}$
- $(x^{(i+1)}, u^{(i+1)}) = \begin{cases} (x_L, u_L) & \text{if } v < \mathcal{A} \\ (x^{(i)}, u^*) & \text{else} \end{cases}$

  with $\Delta(x) = \frac{\partial}{\partial x} logp(x)$,

  and $\mathcal{A} = min\left(1, \frac{p(x_L)}{p(x^{(i)})} exp(-\frac{1}{2}(u_L^T u_L - u^{*T} u^*))\right)$

Slice sampling

Idea: use auxiliary variable $u \in \mathbb{R}$ and extended target distribution

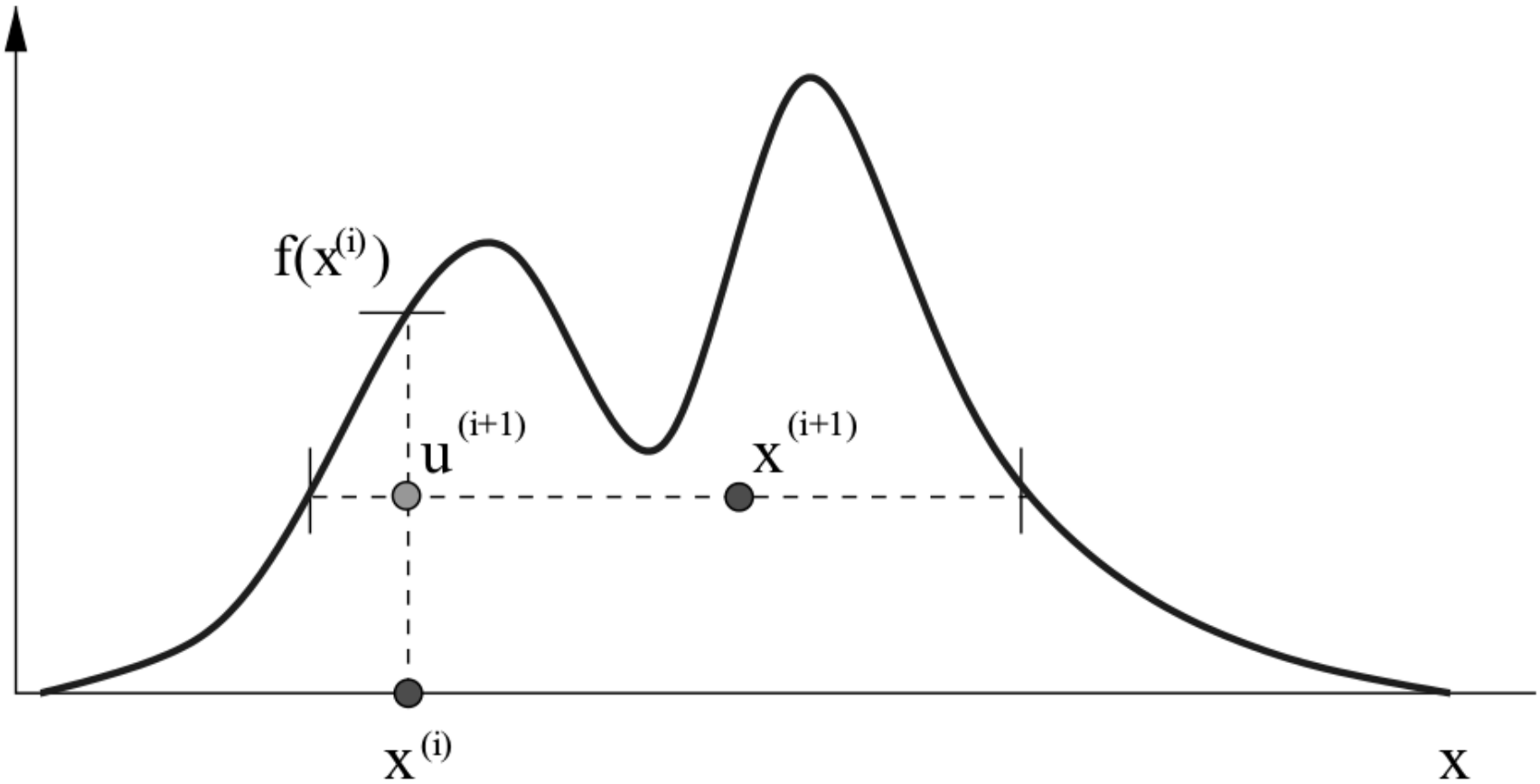
$$p^*(x,u) = \begin{cases} 1 & \text{if } 0 \leq u \leq p(x) \\ 0 & \text{else} \end{cases}$$

$$\text{with } \int p^*(x,u)\,du = \int_0^{p(x)} du = p(x)$$

So we can sample over $p^*(x,u)$ and then ignore u. We can also extend this to L many variables, resulting to following sampler:

- for $l = 1$ to L:
  - sample $u_l^{(i)} \sim \mathcal{U}_{[0,f_l(x^{(i-1)})]}(u_l)$
- sample $x^{(i)} \sim \mathcal{U}_{\mathcal{A}^{(i)}}(x)$
  with $\mathcal{A}^{(i)} = \{x | f_l(x) \geq u_l^{(i)}, l = 1, ..., L\}$

Slice sampling



Thanks for your Attention :-)

Thanks for your attention :-)

References


Andrieu, Christophe and de Freitas, Nando and Doucet, Arnaud and Jordan, Michael I.:
An Introduction to MCMC for Machine Learning
In: *Machine Learning, Kluwer Academic Publishers, 2003, 50, 5-43.*


Murray, Iain:
Markov chain Monte Carlo
In: *Tutorial at Machine Learning Summer School, 2009.*