Sampling and
Markov Chain Monte Carlo

Zoubin Ghahramani

http://learning.eng.cam.ac.uk/zoubin/

zoubin@cs.cmu.edu

Statistical Machine Learning
CMU 10-702 / 36-702
Spring 2008

The role of integration in statistical modelling

- E step of the EM algorithm requires expected sufficient statistics:

$$E[s(h,v)|\theta] = \int s(h,v)\, p(h|v,\theta)\, dh$$

- Bayesian prediction:

$$p(x|\mathcal{D},m) = \int p(x|\theta,\mathcal{D},m)\, p(\theta|\mathcal{D},m)\, d\theta$$

- Computing model evidence (marginal likelihood) for model comparison:

$$p(\mathcal{D}|m) = \int p(\mathcal{D}|\theta,m)\, p(\theta|m)\, d\theta$$

Note that almost every thing I say about integration will also hold for summation.

Examples of Intractability

- Bayesian marginal likelihood/model evidence for Mixture of Gaussians: exact computations are exponential in number of data points

$$p(\mathbf{y}_1, \ldots, \mathbf{y}_N) = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_N} \int p(\theta) \prod_{i=1}^{N} p(\mathbf{y}_i | s_i, \theta) p(s_i | \theta) d\theta$$

- Computing the conditional probability of a variable in a very large multiply connected directed graphical model:

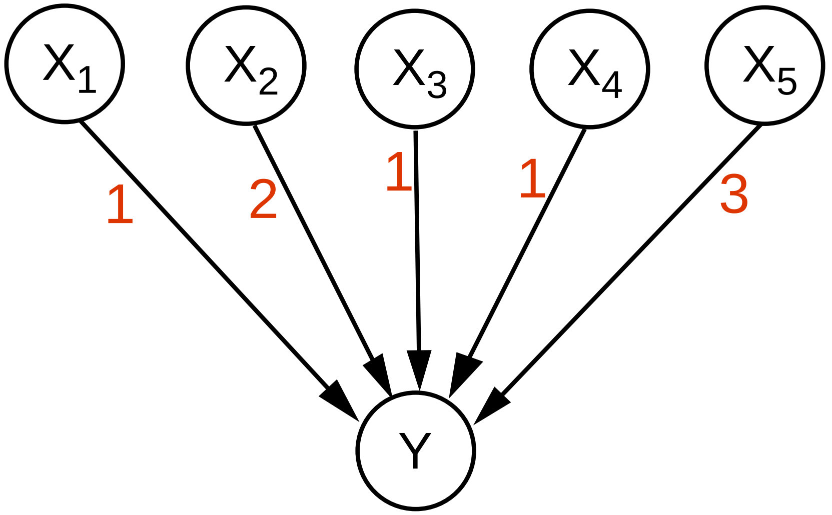
$$p(x_i | X_j = a) = \sum_{\text{all settings of } \mathbf{x} \backslash \{i,j\}} p(x_i, \mathbf{x}, X_j = a) / p(X_j = a)$$

- Computing the hidden state distribution in a general nonlinear dynamical system

$$p(\mathbf{x}_t | \mathbf{y}_1, \ldots, \mathbf{y}_T) \propto \int p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_{t-1} | \mathbf{y}_1, \ldots \mathbf{y}_{t-1}) p(\mathbf{y}_{t+1}, \ldots \mathbf{y}_t | \mathbf{x}_t) d\mathbf{x}_{t-1}$$

Examples of Intractability

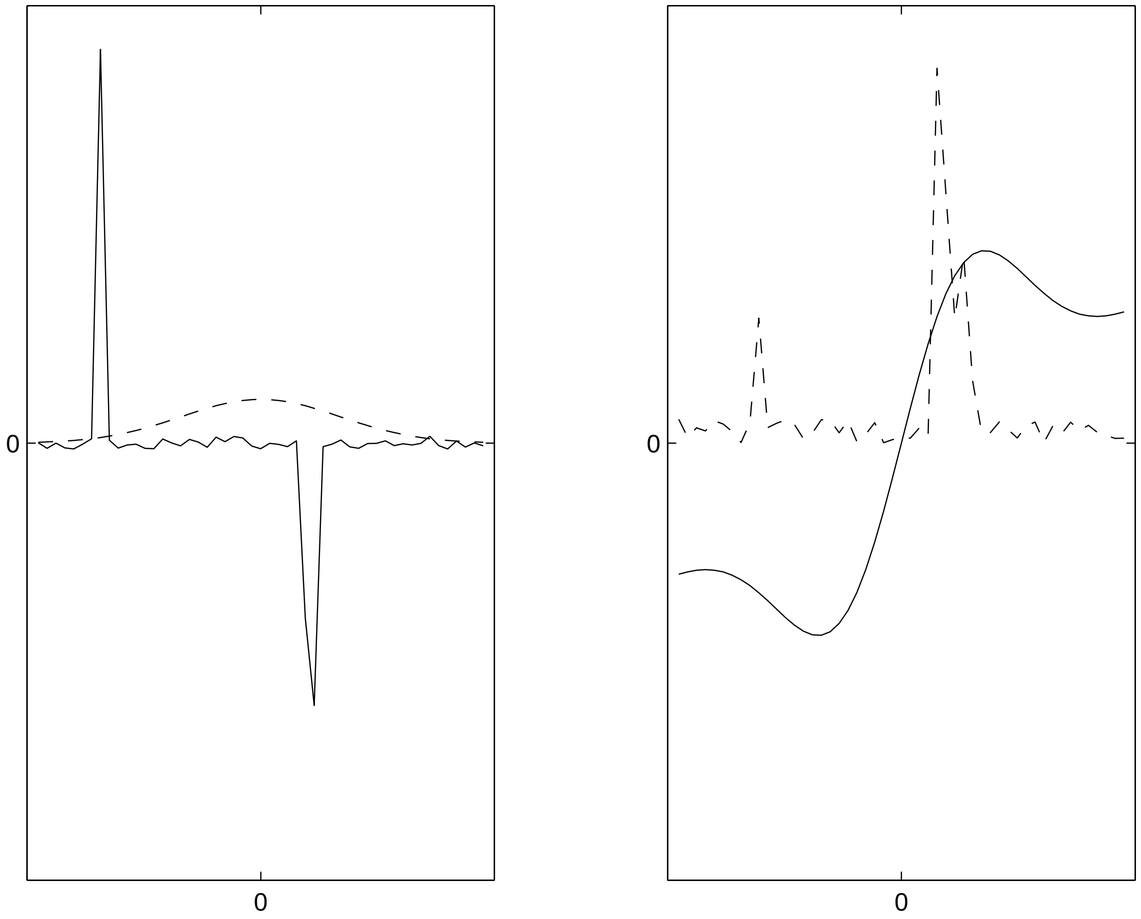
- Multiple cause models:



Y = X1 + 2 X2 + X3 + X4 + 3 X5

Assume X_i are binary, and hidden.
Consider $P(X_1, \ldots, X_5 | Y = 5)$.
What happens if we have more Xs? How is this related to EM?

The integration problem



We often need to compute integrals of the form

$$\int F(x)\, p(x) dx,$$

where $F(x)$ is some function of a random variable X which has probability density $p(x)$.

Three typical difficulties:
left panel: full line is some complicated function, dashed is density;
right panel: full line is some function and dashed is complicated density;
not shown: integral (or sum) in very high dimensions

Sampling Methods

The basic idea of sampling methods is to approximate an intractable integral or sum using samples from some distribution.

Simple Monte Carlo Sampling

Idea: Sample from $p(x)$, average values of $F(x)$.

Simple Monte Carlo:

$$\bar{F} = \int F(x)p(x)dx \simeq \hat{F} = \frac{1}{T}\sum_{t=1}^{T} F(x^{(t)}),$$

where $x^{(t)}$ are (independent) samples drawn from $p(x)$.

Attractions:

- unbiased: $E(\hat{F}) = \bar{F}$
- variance $V(\hat{F})$ goes as $1/T$, independent of dimension!
  $V(\hat{F}) = V(F)/T$ where $V(F) = \int (F(x) - \bar{F})^2 p(x)dx$.

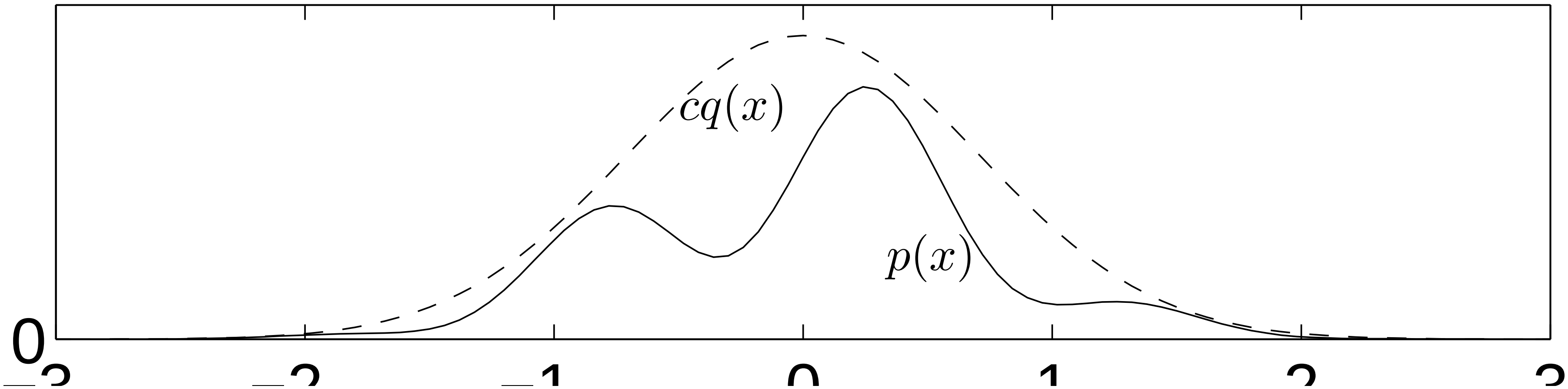
Problems:

- it may be difficult (impossible) to obtain the samples directly from $p(x)$
- regions of high density $p(x)$ may not be regions where $F(x)$ varies a lot

Rejection Sampling

Idea: sample from an upper bound on $p(x)$, rejecting some samples.

- Find a distribution $q(x)$ and a constant c such that $\forall x,\ p(x) \leq cq(x)$.
- Sample x^* from $q(x)$ and accept x^* with probability $p(x^*)/(cq(x^*))$.
- Use accepted points as in simple Monte Carlo: $\sum_{t=1}^{T} F(x^{(t)})$



Problem: often difficult to find $q(x)$ with a small c which is easy to sample from.

Examples:

- Compute $P(X_i = b | X_j = a)$ in a directed graphical model: sample from $P(X)$, reject if $X_j \neq a$, averaging the indicator function $I(X_i = b)$
- Compute $E(x^2 | x > 4)$ for $x \sim \mathcal{N}(0, 1)$

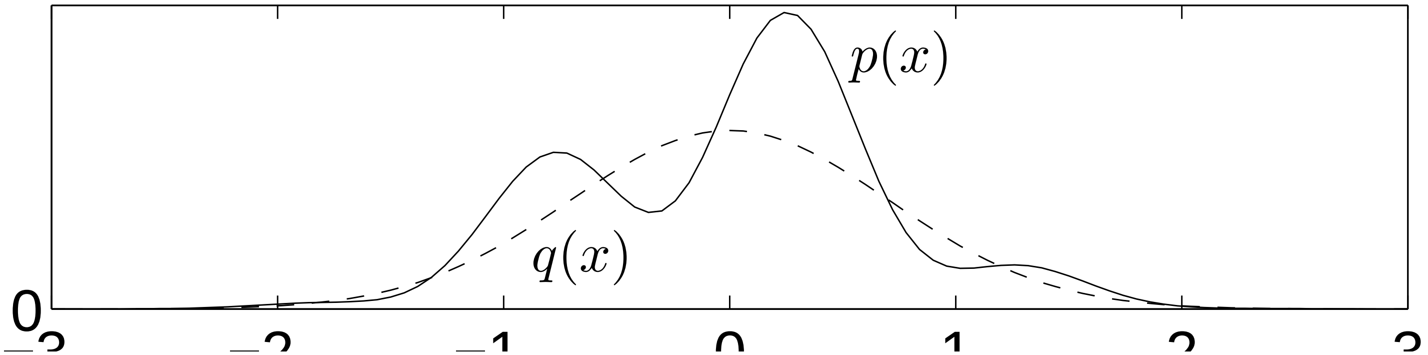
Importance Sampling

Idea: Sample from a different distribution $q(x)$ and weight samples by $p(x)/q(x)$

Sample $x^{(t)}$ from $q(x)$:

$$\int F(x)p(x)dx = \int F(x)\frac{p(x)}{q(x)}q(x)dx \simeq \frac{1}{T}\sum_{t=1}^{T} F(x^{(t)})\frac{p(x^{(t)})}{q(x^{(t)})},$$

where $q(x)$ is non-zero wherever $p(x)$ is; weights $w^{(t)} \equiv p(x^{(t)})/q(x^{(t)})$



Attraction: unbiased; no need for upper bound (cf rejection sampling).

Problems: it may be difficult to find a suitable $q(x)$. Monte Carlo average may be dominated by few samples (high variance); or none of the high weight samples may be found!

Analysis of Importance Sampling

Weights:

$$w^{(t)} \equiv \frac{p(x^{(t)})}{q(x^{(t)})}$$

Define weight function $w(x) = p(x)/q(x)$.
Importance sample is unbiased:

$$E_q(w(x)F(x)) = \int q(x)w(x)F(x)dx = \int p(x)F(x)dx$$

$$E_q(w(x)) = \int q(x)w(x)dx = 1$$

The variance of the weights $V(w(x)) = E_q(w(x)^2) - 1$, where:

$$E_q(w(x)^2) = \int \frac{p(x)^2}{q(x)^2}q(x)dx = \int \frac{p(x)^2}{q(x)}dx$$

Why is high variance of the weights a bad thing?
How does it relate to *effective number of samples*?
What happens if $p(x) = \mathcal{N}(0, \sigma_p^2)$ and $q(x) = \mathcal{N}(0, \sigma_q^2)$?

Markov chain Monte Carlo (MCMC) methods

Assume we are interested in drawing samples from some desired distribution $p^*(x)$.

We define a Markov chain:

$$x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow x_5 \dots$$

where $x_0 \sim p_0(x)$, $x_1 \sim p_1(x)$, etc, with the property that:

$$p_t(x') = \sum_x p_{t-1}(x) T(x \rightarrow x')$$

where $T(x \rightarrow x') = p(X_t = x' | X_{t-1} = x)$ is the **Markov chain transition probability** from x to x'.

We say that $p^*(x)$ is an **invariant (or stationary) distribution** of the Markov chain defined by T iff:

$$p^*(x') = \sum_x p^*(x) T(x \rightarrow x') \qquad \forall x'$$

Markov chain Monte Carlo (MCMC) methods

We have a Markov chain $x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow \ldots$ where $x_0 \sim p_0(x)$, $x_1 \sim p_1(x)$, etc, with the property that:

$$p_t(x') = \sum_x p_{t-1}(x) T(x \rightarrow x')$$

where $T(x \rightarrow x')$ is the Markov chain transition probability from x to x'.

A useful condition that implies invariance of $p^*(x)$ is **detailed balance**:

$$p^*(x')T(x' \rightarrow x) = p^*(x)T(x \rightarrow x')$$

We wish to find **ergodic** Markov chains, which converge to a unique stationary distribution (also called an *equilibrium distribution*) regardless of the initial conditions $p_0(x)$:

$$\lim_{t \rightarrow \infty} p_t(x) = p^*(x)$$

A sufficient condition for the Markov chain to be ergodic is that

$$T^k(x \rightarrow x') > 0 \text{ for all } x \text{ and } x' \text{ where } p^*(x') > 0.$$

That is, if the equilibrium distribution gives non-zero probability to state x', then the Markov chain should be able to reach x' from any x after some finite number of steps, k.

An Overview of Sampling Methods

Monte Carlo Methods:

- Simple Monte Carlo Sampling
- Rejection Sampling
- Importance Sampling
- etc.

Markov Chain Monte Carlo Methods:

- Gibbs Sampling
- Metropolis Algorithm
- Hybrid Monte Carlo
- etc.

Gibbs Sampling

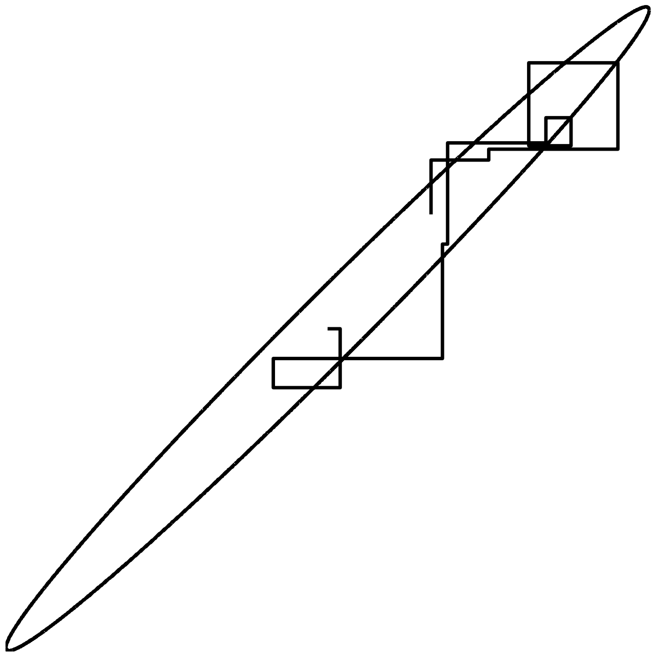
A method for sampling from a multivariate distribution, $p(\mathbf{x})$
Idea: sample from conditional of each var. given the settings of the other variables.

Repeatedly:
1) pick i (either at random or in turn)
2) replace x_i by a sample from the conditional:

$$x_i \sim p(x_i | x_1, \ldots, x_{i-1}, x_{i+1}, \ldots x_n)$$

Gibbs sampling is feasible if it is easy to sample from the conditional probabilities.
This creates a Markov chain

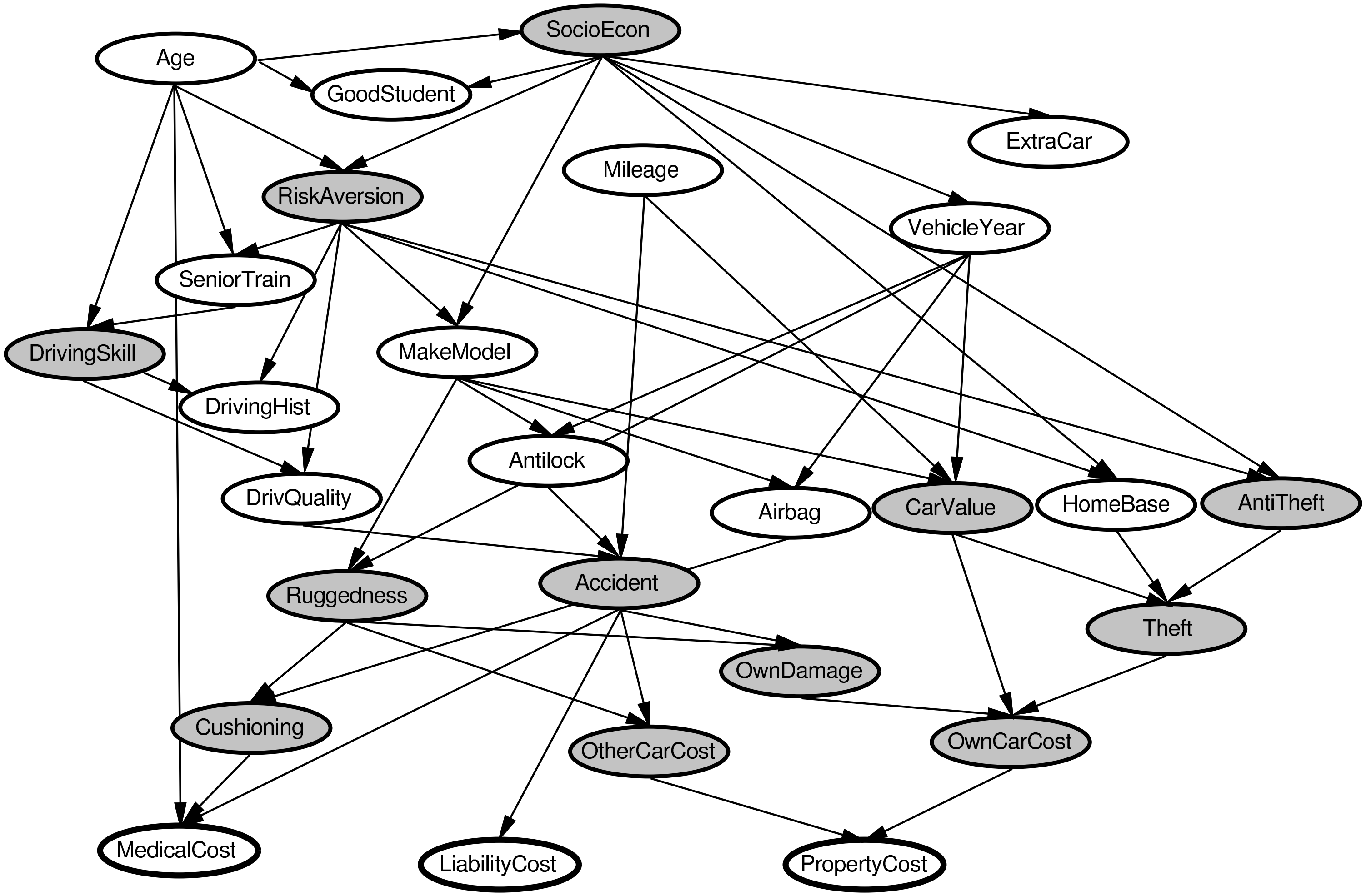
$$\mathbf{x}^{(1)} \rightarrow \mathbf{x}^{(2)} \rightarrow \mathbf{x}^{(3)} \rightarrow \ldots$$



Example: 20 (half-) iterations of Gibbs sampling on a bivariate Gaussian

Under some (mild) conditions, the **equilibium distribution**, i.e. $\lim_{t\rightarrow\infty} p(\mathbf{x}^{(t)})$, of this Markov chain is $p(\mathbf{x})$ (demo)

Gibbs Sampling in Graphical Models



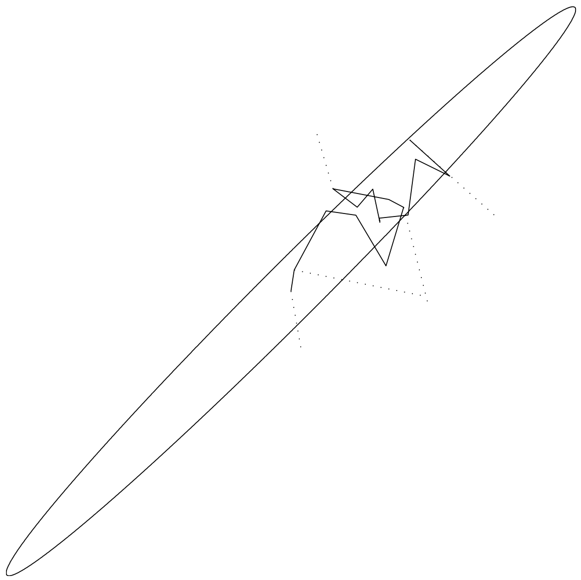
Initialize all variables to some settings.
Sample each variable conditional on other variables.

The BUGS software implements this algorithm for a variety of graphical models.

The Metropolis algorithm

Idea: Propose a change to current state; accept or reject.

Each step: Starting from the current state $\mathbf{x}$,



1. Propose a new state $\mathbf{x}'$ from a distribution $S(\mathbf{x}'|\mathbf{x})$.
2. Accept the new state with probability $\min(1, \frac{p(\mathbf{x}')S(\mathbf{x}|\mathbf{x}')}{p(\mathbf{x})S(\mathbf{x}'|\mathbf{x})})$;
3. Otherwise retain the old state.

Example: 20 iterations of global metropolis sampling from bivariate Gaussian; rejected proposals are dotted.

- Symmetric proposals, $S(\mathbf{x}'|\mathbf{x}) = S(\mathbf{x}|\mathbf{x}')$, even simpler.
- **Local** (changing one x_i) vs **global** (changing all $\mathbf{x}$) proposal distributions.
- Note, we need only to compute ratios of probabilities (no normalizing constants).

(demo)

Hybrid Monte Carlo: overview

Motivation: The typical distance traveled by a random walk in n steps is proportional to $\sqrt{n}$. We want to seek regions of high probability while avoiding random walk behavior.

Assume that we wish to sample from $p(\mathbf{x})$ while avoiding random walk behaviour. If we can compute derivatives of $p(\mathbf{x})$ with respect to $\mathbf{x}$, this is *useful information* and we should be able to use it to draw samples better.

Hybrid Monte Carlo: We think of a fictitious physical system with a particle which has position $\mathbf{x}$ and momentum $\mathbf{v}$. We will design a sampler which avoids random walks in $\mathbf{x}$ by simulating a dynamical system.

We simulate the dynamical system in such a way that the marginal distribution of positions, $p(\mathbf{x})$, ignoring the momentum variables, corresponds to the desired distribution.

Hybrid Monte Carlo: the dynamical system

In the physical system, positions $\mathbf{x}$ corresponding to random variables of interest are augmented by momentum variables $\mathbf{v}$:

$$p(\mathbf{x},\mathbf{v}) \propto \exp(-H(\mathbf{x},\mathbf{v})) \qquad H(\mathbf{x},\mathbf{v}) = E(\mathbf{x}) + K(\mathbf{v})$$
$$E(\mathbf{x}) = -\log p(\mathbf{x}) \qquad K(\mathbf{v}) = \tfrac{1}{2}\sum_i v_i^2$$

Importantly, note $\int p(\mathbf{x},\mathbf{v}) d\mathbf{v} = p(\mathbf{x})$ and $p(\mathbf{v}) = N(0,I)$.

We think of $E(\mathbf{x})$ as the potential energy of being in state $\mathbf{x}$, and $K(\mathbf{v})$ as the kinetic energy associated with momentum $\mathbf{v}$. We assume "mass" $= 1$, so momentum=velocity.

The system evolves at constant total energy H according to Hamiltonian dynamics:

$$\frac{dx_i}{dt} = \frac{\partial H}{\partial v_i} = v_i \qquad \frac{dv_i}{dt} = -\frac{\partial H}{\partial x_i} = -\frac{\partial E}{\partial x_i}.$$

The first equation says derivative of position is velocity. The second equation says that the system accelerates in the direction that decreases potential energy.

Think of a ball rolling on a frictionless hilly surface.

Hybrid Monte Carlo: how to simulate the dynamical system

We can simulate the above differential equations by discretising time and running some difference equations on a computer. This introduces hopefully small errors. (The errors we care about are errors which change the total energy—we will correct for these by occasionally rejecting moves that change the energy.)

A good way to simulate this is using leapfrog simulation. We take L discrete steps of size ϵ to simulate the system evolving for $L\epsilon$ time:

$$\begin{aligned}
\hat{v}_i(t+\frac{\epsilon}{2}) &= \hat{v}_i(t) - \frac{\epsilon}{2}\,\frac{\partial E(\hat{x}(t))}{\partial x_i} \\
\hat{x}_i(t+\epsilon) &= \hat{x}_i(t) + \epsilon\frac{\hat{v}_i(t+\frac{\epsilon}{2})}{m_i} \\
\hat{v}_i(t+\epsilon) &= \hat{v}_i(t+\frac{\epsilon}{2}) - \frac{\epsilon}{2}\,\frac{\partial E(\hat{x}(t+\epsilon))}{\partial x_i}
\end{aligned}$$

Hybrid Monte Carlo: properties of the dynamical system

Hamiltonian dynamics has the following important properties:

1) preserves total energy, H,
2) is reversible in time
3) preserves phase space volumes (Liouville's theorem)

The leapfrog discretisation only approximately preserves the total energy H, and

1) is reversible in time
2) preserves phase space volume

The dynamical system is simulated using the leapfrog discretisation and the new state is used as a proposal in the Metropolis algorithm to eliminate the bias caused by the leapfrog approximation

Hybrid Monte Carlo Algorithm

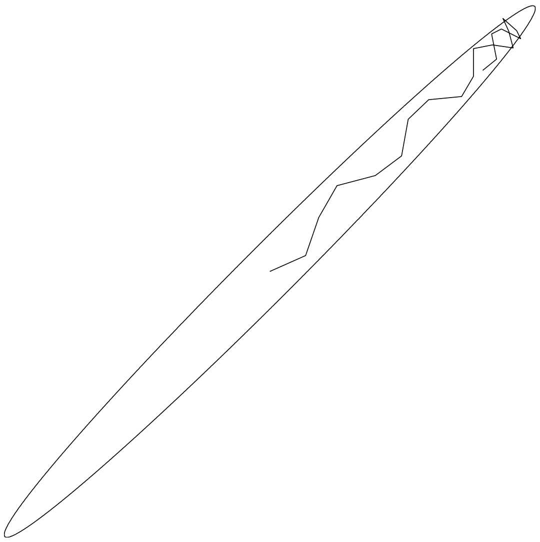
1) A new state is proposed by deterministically simulating a trajectory with L discrete steps from $(\mathbf{x}, \mathbf{v})$ to $(\mathbf{x}^*, \mathbf{v}^*)$. The new state $(\mathbf{x}^*, \mathbf{v}^*)$ is accepted with probability:

$$\min(1, \exp(-(H(\mathbf{v}^*, \mathbf{x}^*) - H(\mathbf{v}, \mathbf{x})))),$$

otherwise the state remains the same.

2) Stochastically update the momenta using Gibbs sampling

$$\mathbf{v} \sim p(\mathbf{v}|\mathbf{x}) = p(\mathbf{v}) = N(0, I)$$



Example: $L = 20$ leapfrog iterations when sampling from a bivariate Gaussian

(demo)

Summary

Rejection and Importance sampling are based on independent samples from $q(\mathbf{x})$.

These methods are not suitable for high dimensional problems.

The Metropolis method, does not give independent samples, but can be used successfully in high dimensions.

Gibbs sampling is used extensively in practice.

- parameter-free
- requires simple conditional distributions

Hybrid Monte Carlo is used extensively in continuous systems

- avoids random walks
- requires setting of ε and L.

Monte Carlo in Practice

Although very general, and can be applied in systems with 1000's of variables.

Care must be taken when using MCMC

- high dependency between variables may cause slow exploration
  - how long does 'burn-in' take?
  - how correlated are the samples generated?
  - slow exploration may be hard to detect – but you might not be getting the right answer
- diagnosing a convergence problem may be difficult
- curing a convergence problem may be even more difficult