Markov chain Monte Carlo

Machine Learning Summer School 2009

http://mlg.eng.cam.ac.uk/mlss09/

Lecture 1 and some slides from lecture 2

Iain Murray

http://www.cs.toronto.edu/~murray/

A statistical problem

What is the average height of the MLSS lecturers?

Method: measure their heights, add them up and divide by $N = 20$.

What is the average height f of people p in Cambridge $\mathcal{C}$?

$$E_{p\in\mathcal{C}}[f(p)] \equiv \frac{1}{|\mathcal{C}|}\sum_{p\in\mathcal{C}} f(p), \quad \text{“intractable”?}$$

$$\approx \frac{1}{S}\sum_{s=1}^{S} f\big(p^{(s)}\big), \quad \text{for random survey of } S \text{ people } \{p^{(s)}\} \in \mathcal{C}$$

Surveying works for large and notionally infinite populations.

Simple Monte Carlo

Statistical sampling can be applied to any expectation:

In general:

$$\int f(x) P(x) \, \mathrm{d}x \; \approx \; \frac{1}{S} \sum_{s=1}^{S} f(x^{(s)}), \quad x^{(s)} \sim P(x)$$

Example: making predictions

$$\begin{aligned} p(x|\mathcal{D}) \; &= \; \int P(x|\theta, \mathcal{D}) P(\theta|\mathcal{D}) \, \mathrm{d}\theta \\ &\approx \; \frac{1}{S} \sum_{s=1}^{S} P(x|\theta^{(s)}, \mathcal{D}), \quad \theta^{(s)} \sim P(\theta|\mathcal{D}) \end{aligned}$$

Another example: finding the E-step statistics in EM

Properties of Monte Carlo

Estimator: $\displaystyle \int f(x) P(x)\, \mathrm{d}x \;\approx\; \hat{f} \;\equiv\; \frac{1}{S}\sum_{s=1}^{S} f(x^{(s)}), \quad x^{(s)} \sim P(x)$

Estimator is unbiased:

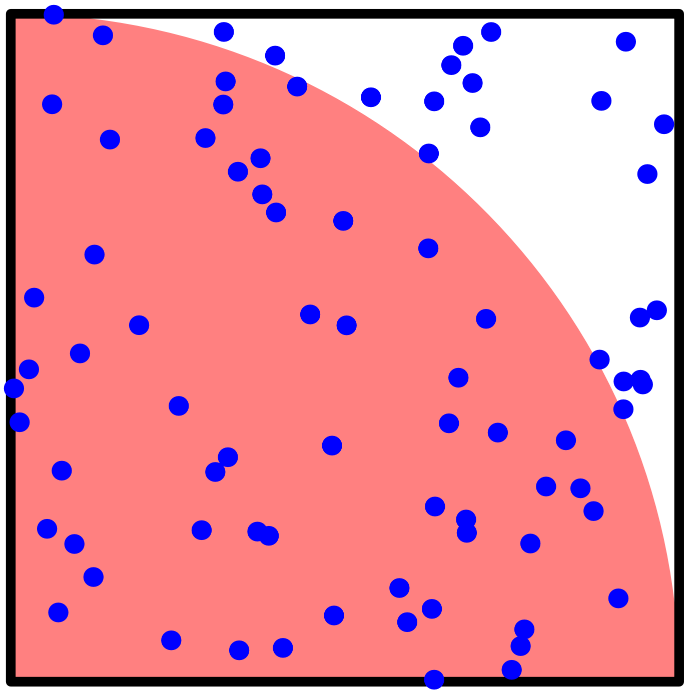
$$\mathbb{E}_{P(\{x^{(s)}\})}\left[\hat{f}\right] \;=\; \frac{1}{S}\sum_{s=1}^{S}\mathbb{E}_{P(x)}[f(x)] \;=\; \mathbb{E}_{P(x)}[f(x)]$$

Variance shrinks $\propto 1/S$:

$$\mathrm{var}_{P(\{x^{(s)}\})}\left[\hat{f}\right] \;=\; \frac{1}{S^2}\sum_{s=1}^{S}\mathrm{var}_{P(x)}[f(x)] \;=\; \mathrm{var}_{P(x)}[f(x)]\,/S$$

"Error bars" shrink like $\sqrt{S}$

A dumb approximation of π



$$P(x,y) = \begin{cases} 1 & 0<x<1 \text{ and } 0<y<1 \\ 0 & \text{otherwise} \end{cases}$$

$$\pi = 4 \iint \mathbb{I}\left((x^2+y^2)<1\right) P(x,y)\, \mathrm{d}x\, \mathrm{d}y$$

```
octave:1> S=12; a=rand(S,2); 4*mean(sum(a.*a,2)<1)
ans = 3.3333
octave:2> S=1e7; a=rand(S,2); 4*mean(sum(a.*a,2)<1)
ans = 3.1418
```

Aside: don't always sample!

"Monte Carlo is an extremely bad method; it should be used only when all alternative methods are worse."

— Alan Sokal, 1996

Example: numerical solutions to 1D integrals are fast

```
octave:1> 4 * quadl(@(x) sqrt(1-x.^2), 0, 1, tolerance)
```

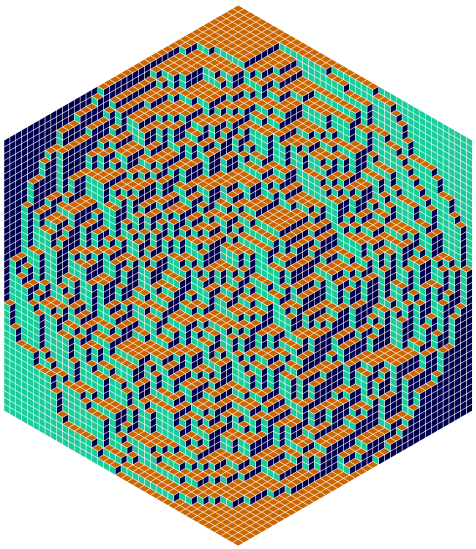
Gives π to 6 dp's in 108 evaluations, machine precision in 2598.

(NB Matlab's `quadl` fails at zero tolerance)

Other lecturers are covering alternatives for higher dimensions.

No approx. integration method always works. Sometimes Monte Carlo is the best.

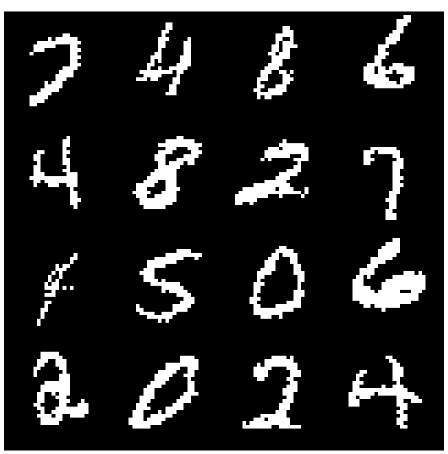
Eye-balling samples



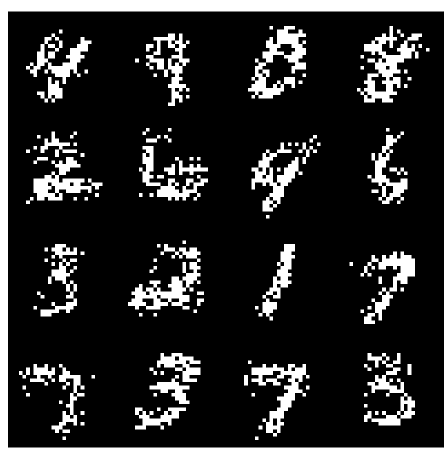
Sometimes samples are pleasing to look at:
(if you're into geometrical combinatorics)

Figure by Propp and Wilson. Source: MacKay textbook.

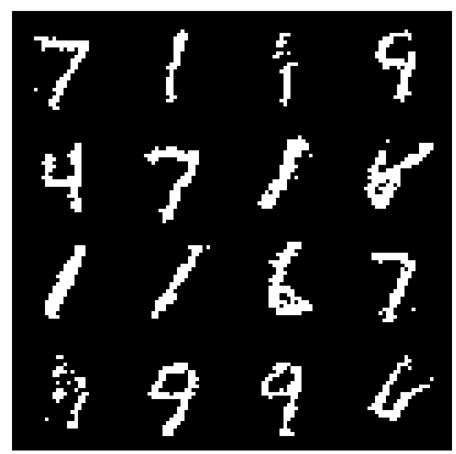
Sanity check probabilistic modelling assumptions:



Data samples

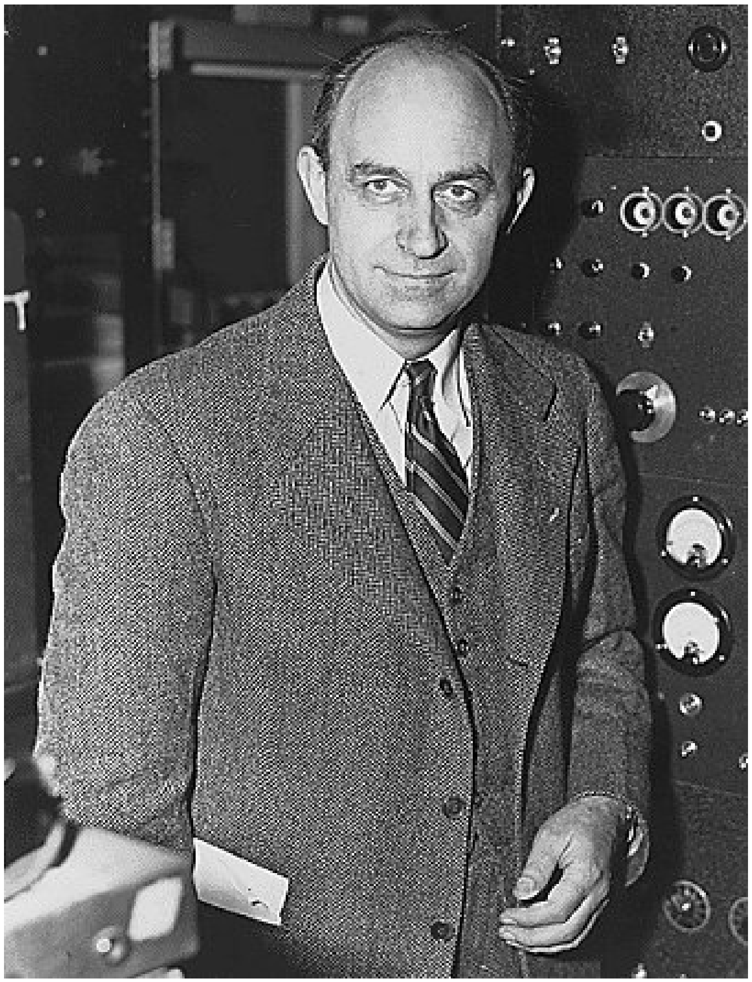


MoB samples



RBM samples

Monte Carlo and Insomnia



Enrico Fermi (1901–1954) took great delight in astonishing his colleagues with his remakably accurate predictions of experimental results. . . he revealed that his "guesses" were really derived from the statistical sampling techniques that he used to calculate with whenever insomnia struck in the wee morning hours!

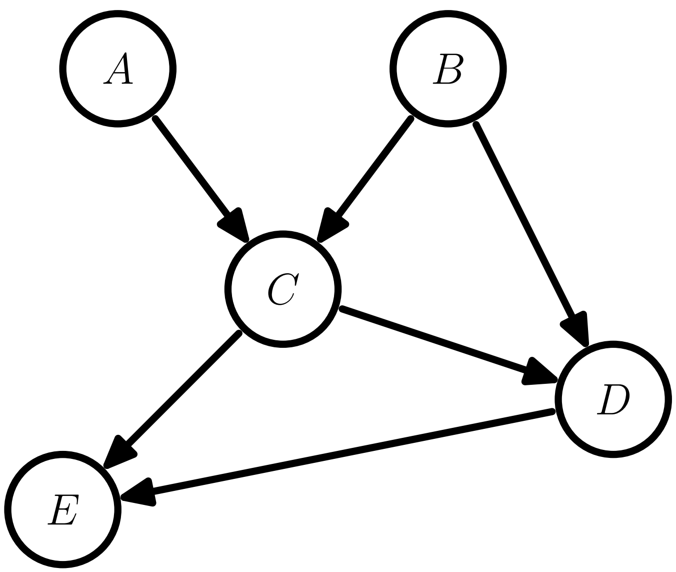
—*The beginning of the Monte Carlo method*, N. Metropolis

Sampling from a Bayes net

Ancestral pass for directed graphical models:

— sample each top level variable from its marginal

— sample each other node from its conditional once its parents have been sampled

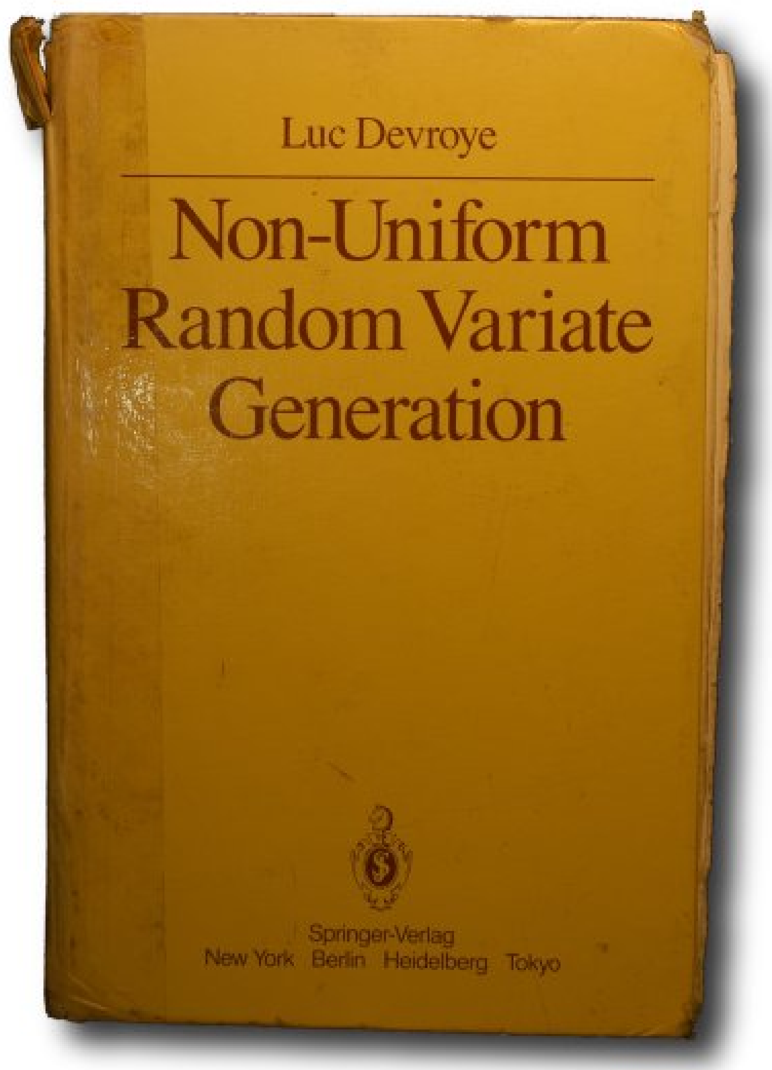


Sample:

$$A \sim P(A)$$
$$B \sim P(B)$$
$$C \sim P(C \,|\, A, B)$$
$$D \sim P(D \,|\, B, C)$$
$$E \sim P(D \,|\, C, D)$$

$$P(A, B, C, D, E) = P(A)\, P(B)\, P(C \,|\, A, B)\, P(D \,|\, B, C)\, P(E \,|\, C, D)$$

Sampling the conditionals



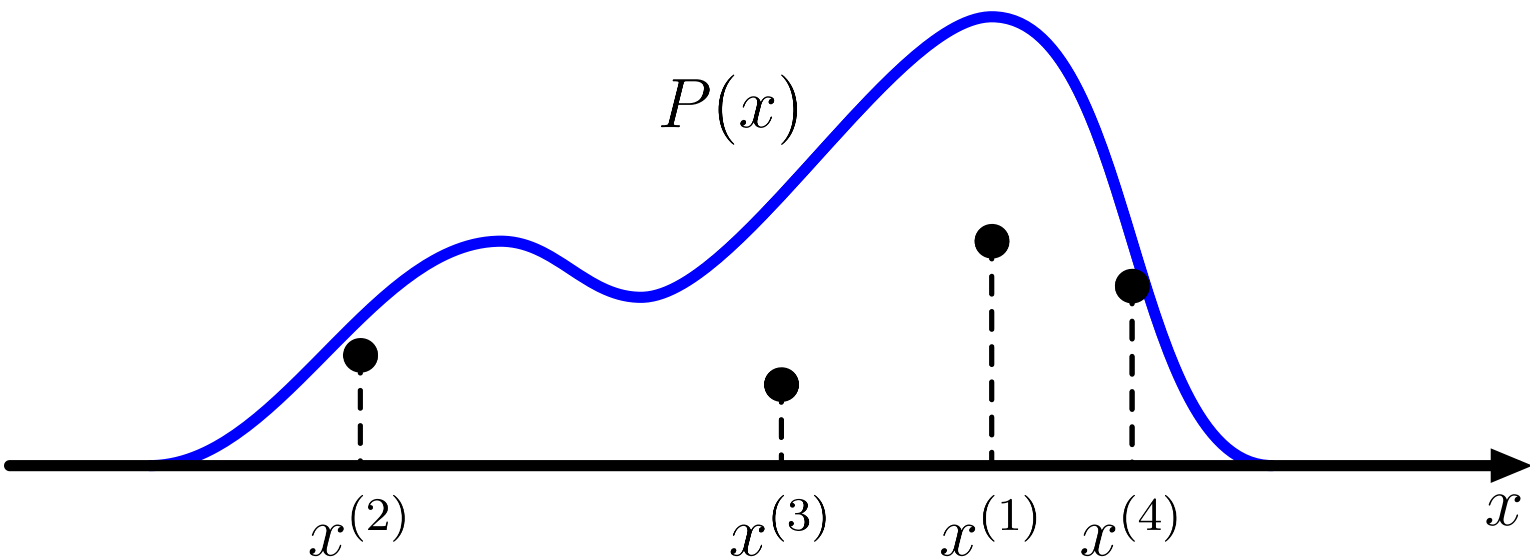
Use library routines for univariate distributions
(and some other special cases)

This book (free online) explains how many of them work

`http://cg.scs.carleton.ca/~luc/rnbookindex.html`

Sampling from distributions

Draw points uniformly under the curve:



Probability mass to left of point $\sim$ Uniform[0,1]

Sampling from distributions

How to convert samples from a Uniform[0,1] generator:

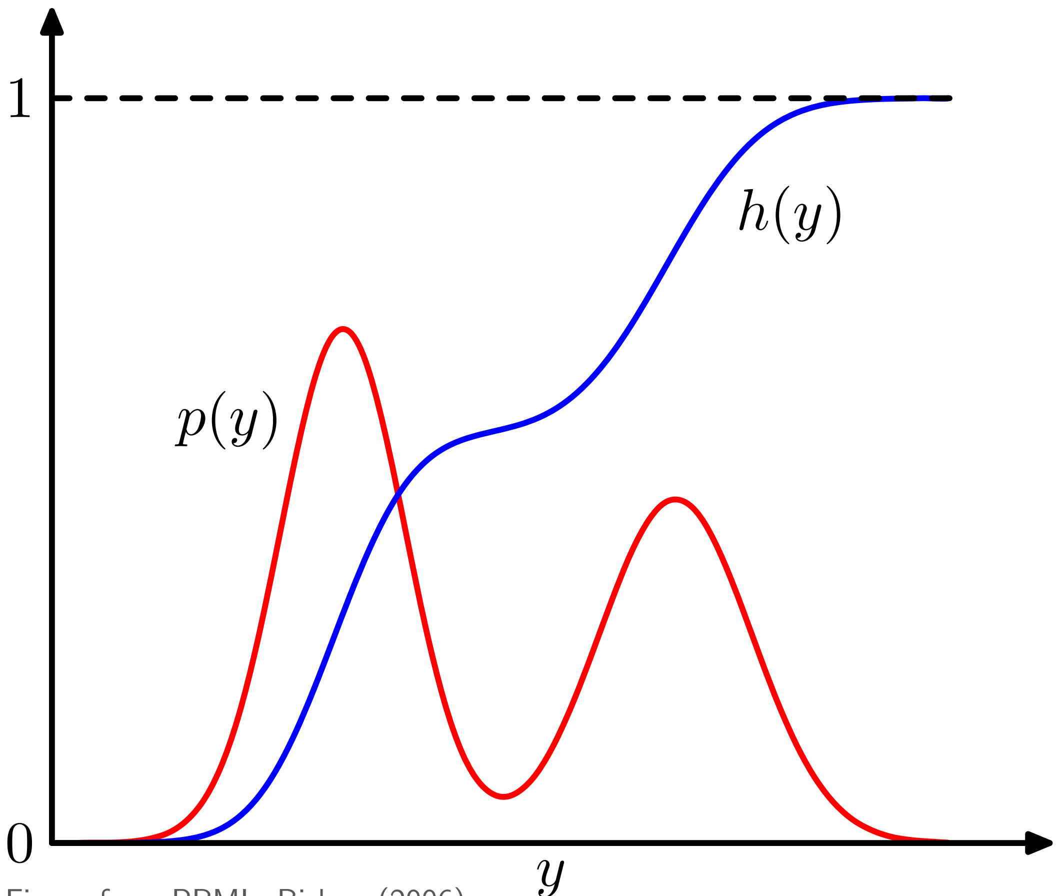


Figure from PRML, Bishop (2006)

$h(y) = \int_{-\infty}^{y} p(y')\,\mathrm{d}y'$

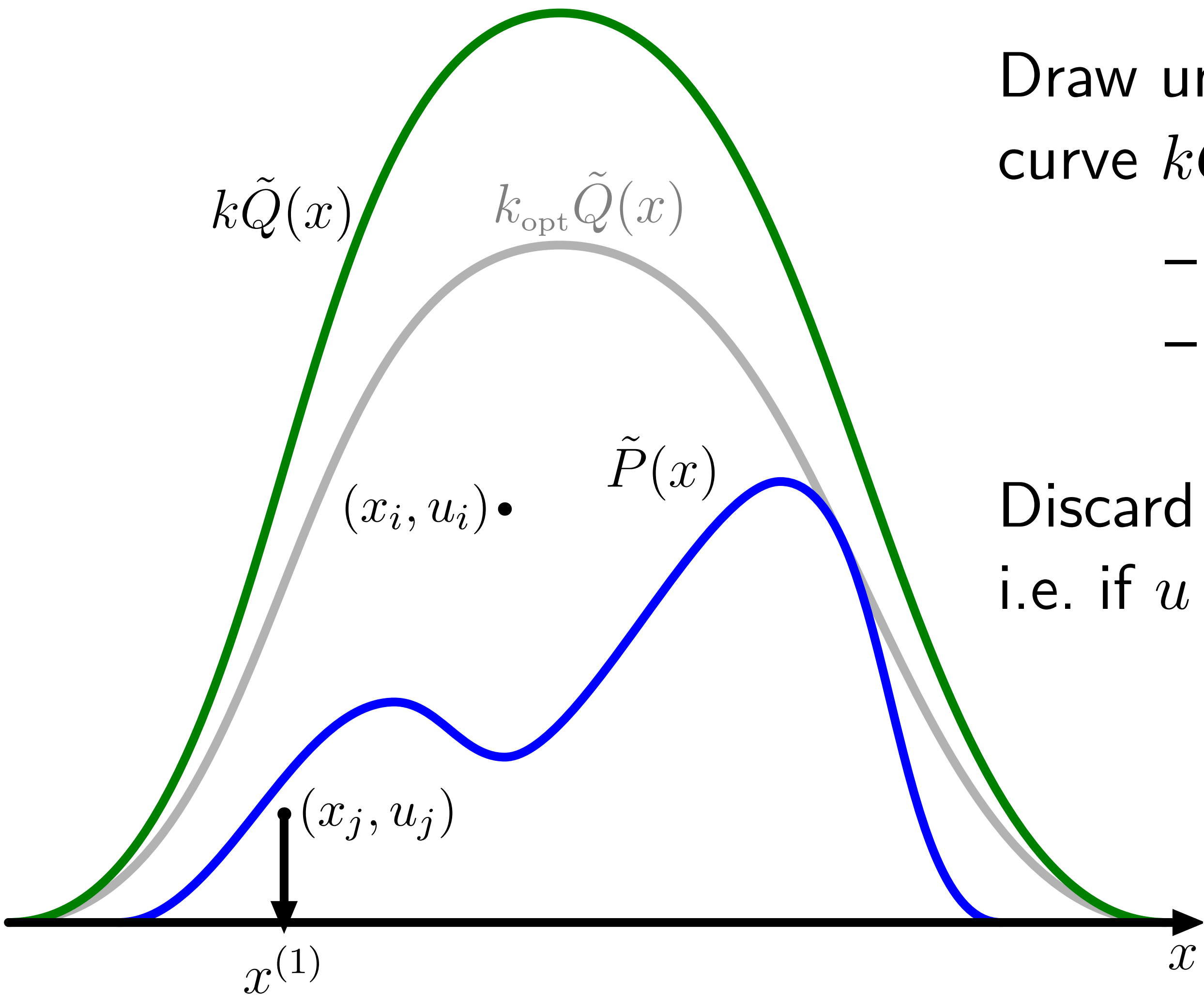
Draw mass to left of point:
$u \sim$ Uniform[0,1]

Sample, $y(u) = h^{-1}(u)$

Although we can't always compute and invert $h(y)$

Rejection sampling

Sampling underneath a $\tilde{P}(x) \propto P(x)$ curve is also valid



Draw underneath a simple curve $k\tilde{Q}(x) \geq \tilde{P}(x)$:

– Draw $x \sim Q(x)$

– height $u \sim \text{Uniform}[0, k\tilde{Q}(x)]$

Discard the point if above $\tilde{P}$, i.e. if $u > \tilde{P}(x)$

Importance sampling

Computing $\tilde{P}(x)$ and $\tilde{Q}(x)$, then *throwing* x *away* seems wasteful
Instead rewrite the integral as an expectation under Q:

$$\int f(x)P(x)\,\mathrm{d}x \;=\; \int f(x)\frac{P(x)}{Q(x)}Q(x)\,\mathrm{d}x, \qquad (Q(x)>0 \text{ if } P(x)>0)$$

$$\approx\; \frac{1}{S}\sum_{s=1}^{S} f(x^{(s)})\frac{P(x^{(s)})}{Q(x^{(s)})}, \quad x^{(s)} \sim Q(x)$$

This is just simple Monte Carlo again, so it is unbiased.

Importance sampling applies when the integral is not an expectation.
Divide and multiply any integrand by a convenient distribution.

Importance sampling (2)

Previous slide assumed we could evaluate $P(x) = \tilde{P}(x)/\mathcal{Z}_P$

$$\int f(x)P(x)\,\mathrm{d}x \;\approx\; \frac{\mathcal{Z}_Q}{\mathcal{Z}_P}\,\frac{1}{S}\sum_{s=1}^{S} f(x^{(s)})\underbrace{\frac{\tilde{P}(x^{(s)})}{\tilde{Q}(x^{(s)})}}_{\tilde{r}^{(s)}}, \quad x^{(s)} \sim Q(x)$$

$$\approx\; \cancel{\frac{1}{S}}\sum_{s=1}^{S} f(x^{(s)})\frac{\tilde{r}^{(s)}}{\cancel{\frac{1}{S}}\sum_{s'}\tilde{r}^{(s')}} \;\equiv\; \sum_{s=1}^{S} f(x^{(s)})w^{(s)}$$

This estimator is **consistent** but **biased**

Exercise: Prove that $\mathcal{Z}_P/\mathcal{Z}_Q \approx \frac{1}{S}\sum_s \tilde{r}^{(s)}$

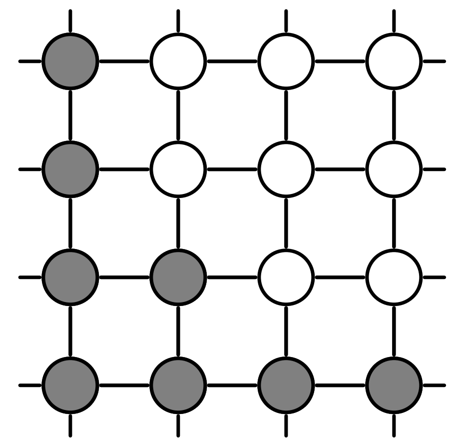
Summary so far

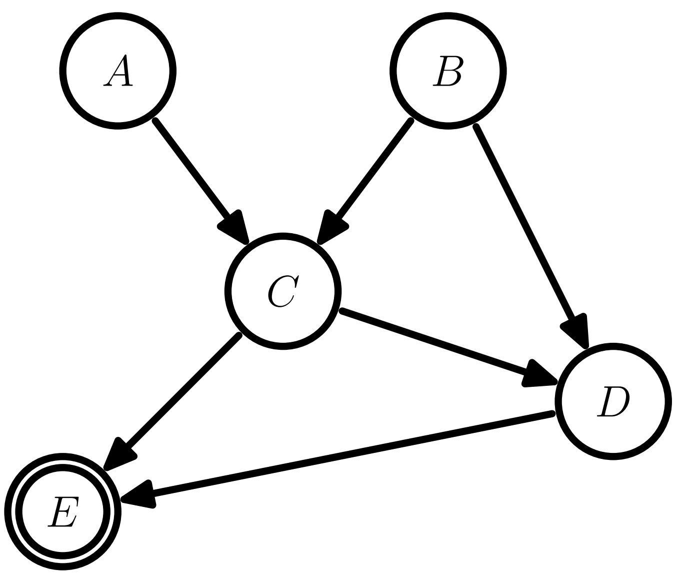
- Sums and integrals, often expectations, occur frequently in statistics
- **Monte Carlo** approximates expectations with a sample average
- **Rejection sampling** draws samples from complex distributions
- **Importance sampling** applies Monte Carlo to any sum/integral

Application to large problems

We often can't decompose $P(X)$ into low-dimensional conditionals

Undirected graphical models: $p(x) = \frac{1}{\mathcal{Z}} \prod_i f_i(x)$





Posterior of a directed graphical model

$$P(A, B, C, D \,|\, E) = \frac{P(A, B, C, D, E)}{P(E)}$$

We usually don't know $\mathcal{Z}$ or $P(E)$

Application to large problems

Rejection & importance sampling scale badly with dimensionality

Example:

$$P(x) = \mathcal{N}(0,\, \mathbb{I}), \quad Q(x) = \mathcal{N}(0,\, \sigma^2\mathbb{I})$$

Rejection sampling:

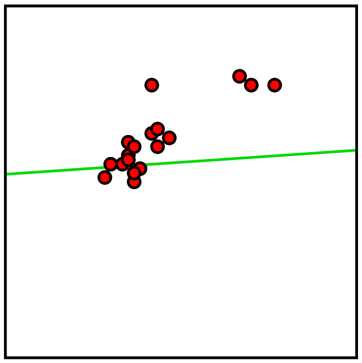
Requires $\sigma > 1$. Fraction of proposals rejected $= \sigma^D$

Importance sampling:

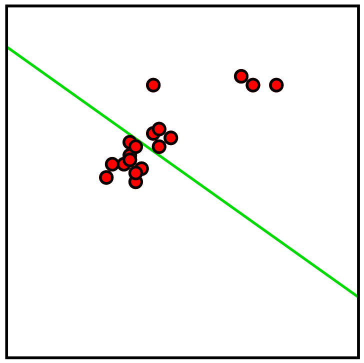
Variance of importance weights $= \sigma^D\, (2\pi^2)^{D/2}/(1-1/2\sigma^2) - 1$

Infinite / undefined variance if $\sigma \leq 1/\sqrt{2}$

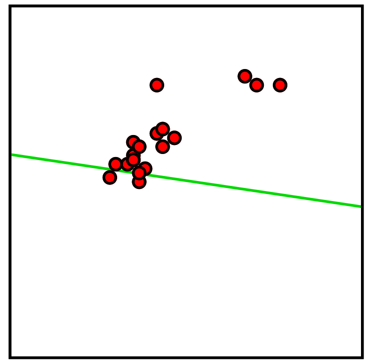
Importance sampling weights



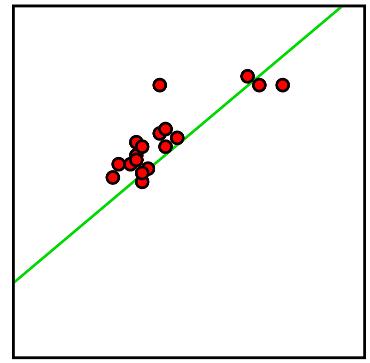
$w = $0.00548



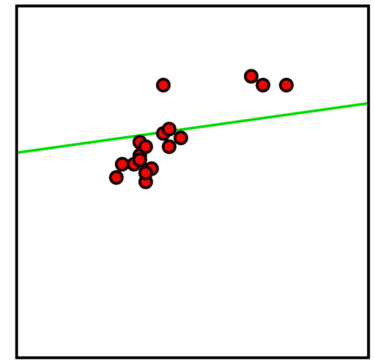
$w = $1.59e-08



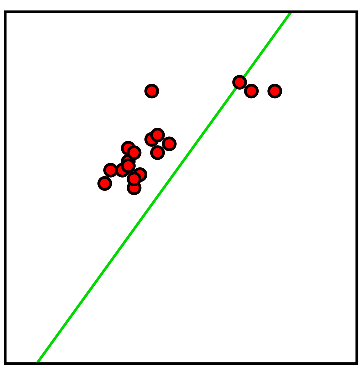
$w = $9.65e-06



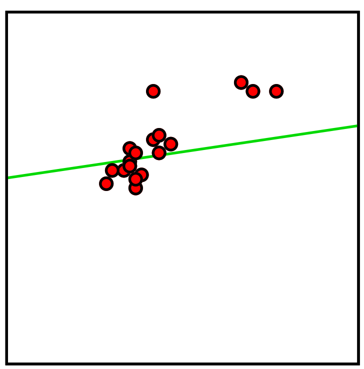
$w = $0.371



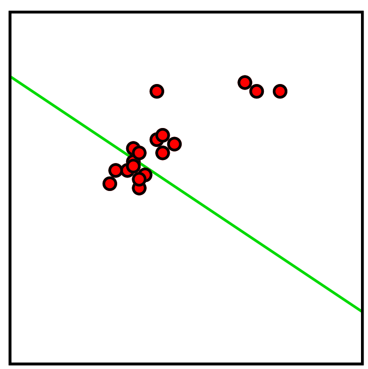
$w = $0.103



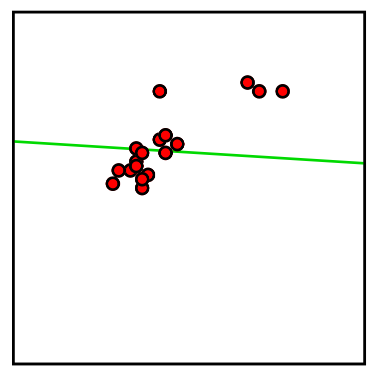
$w = $1.01e-08



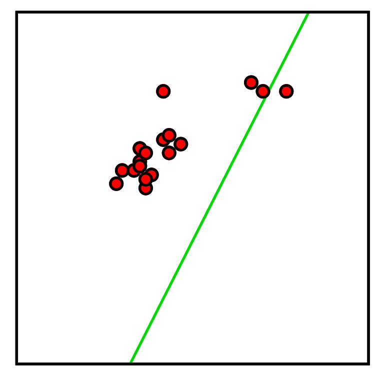
$w = $0.111



$w = $1.92e-09

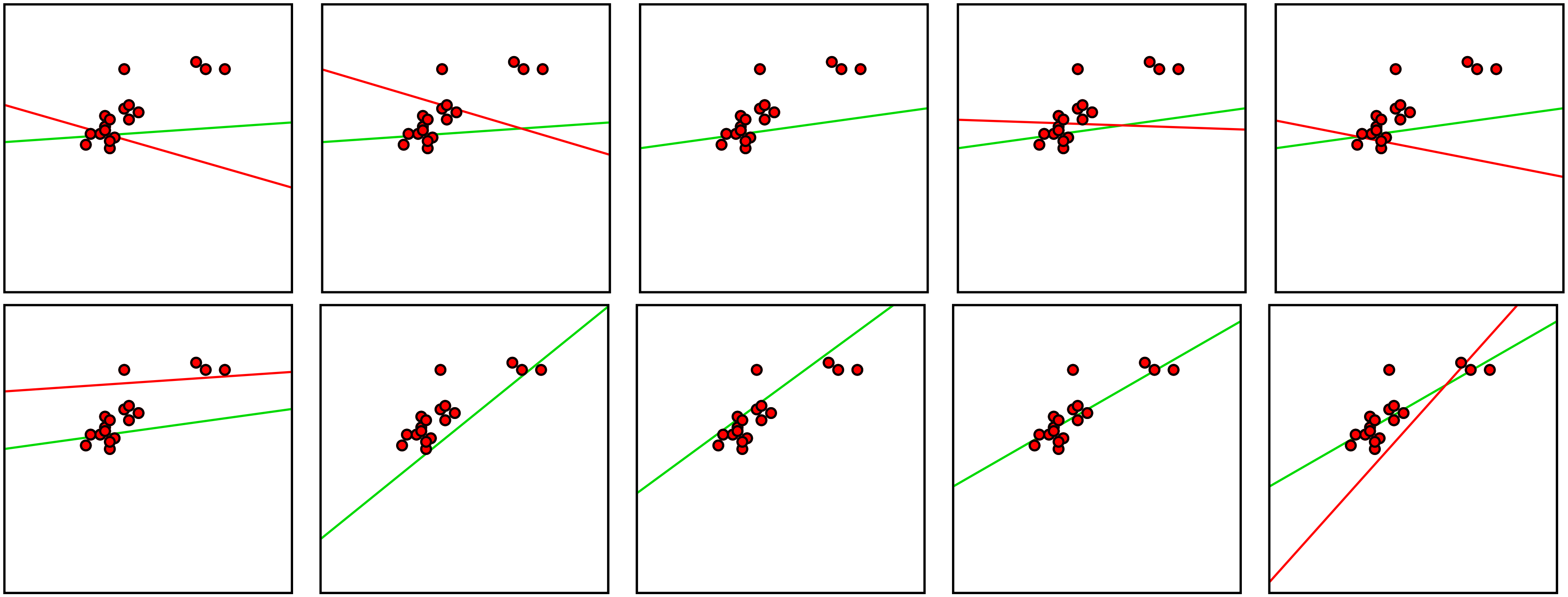


$w = $0.0126

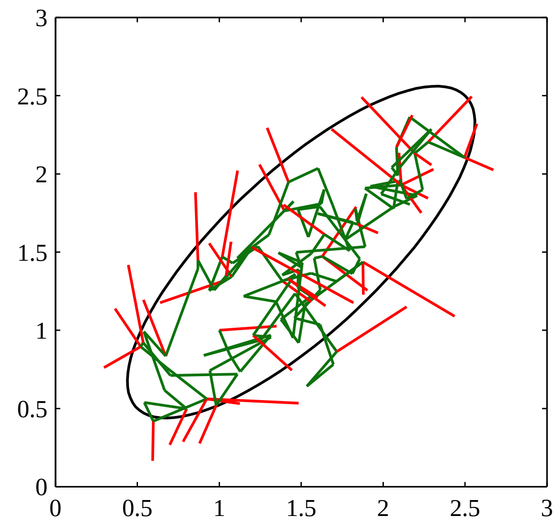


$w = $1.1e-51

Metropolis algorithm



- Perturb parameters: $Q(\theta'; \theta)$, e.g. $\mathcal{N}(\theta, \sigma^2)$
- Accept with probability $\min\left(1, \dfrac{\tilde{P}(\theta'|\mathcal{D})}{\tilde{P}(\theta|\mathcal{D})}\right)$
- Otherwise **keep old parameters**



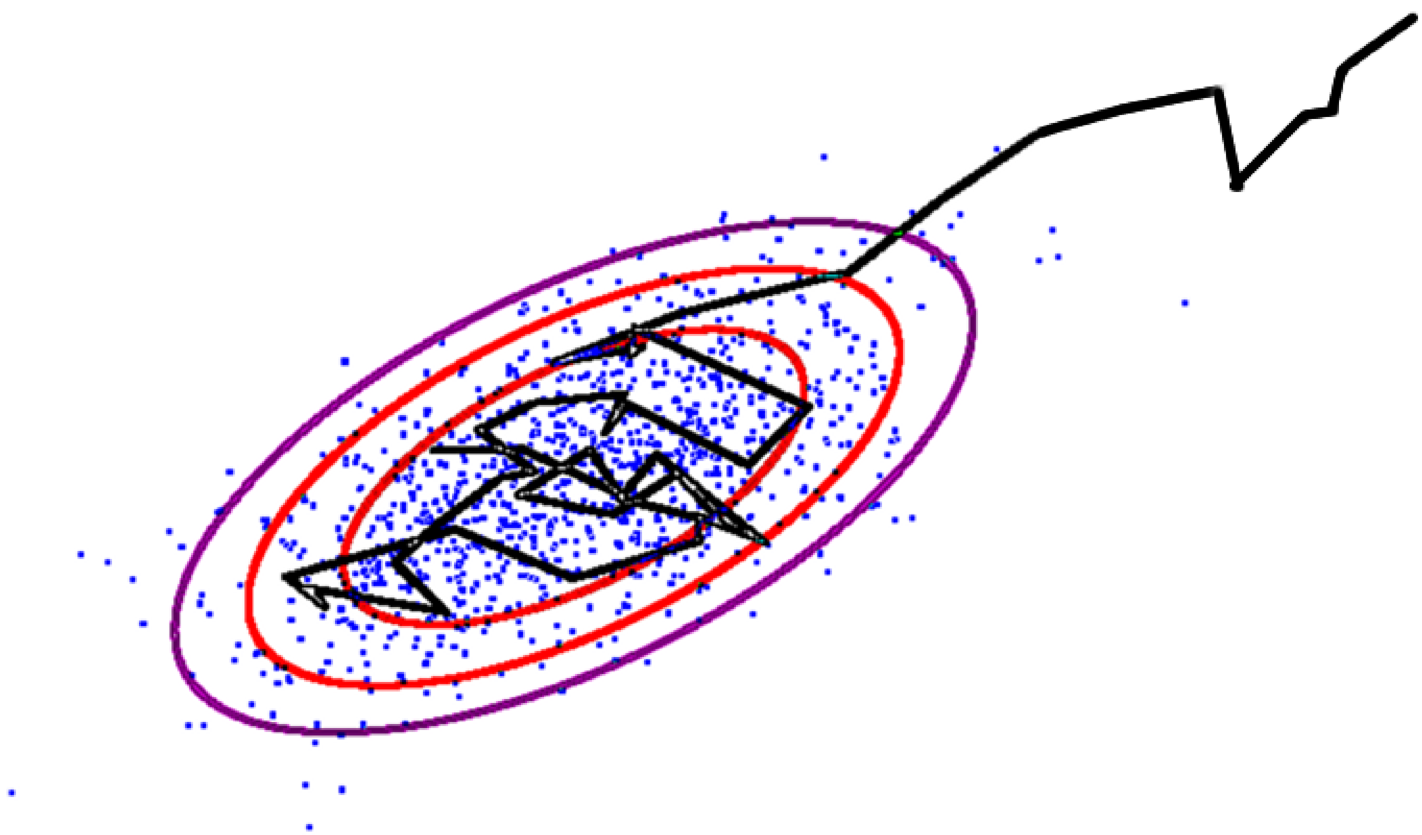
This subfigure from PRML, Bishop (2006)

Detail: Metropolis, as stated, requires $Q(\theta'; \theta) = Q(\theta; \theta')$

Markov chain Monte Carlo

Construct a biased random walk that explores target dist $P^\star(x)$

Markov steps, $x_t \sim T(x_t \leftarrow x_{t-1})$



MCMC gives approximate, correlated samples from $P^\star(x)$

Transition operators

Discrete example

$$P^\star = \begin{pmatrix} 3/5 \\ 1/5 \\ 1/5 \end{pmatrix} \qquad T = \begin{pmatrix} 2/3 & 1/2 & 1/2 \\ 1/6 & 0 & 1/2 \\ 1/6 & 1/2 & 0 \end{pmatrix} \qquad T_{ij} = T(x_i \!\leftarrow\! x_j)$$

$P^\star$ is an **invariant distribution** of T because $TP^\star \!=\! P^\star$, i.e.

$$\sum_x T(x' \!\leftarrow\! x) P^\star(x) = P^\star(x')$$

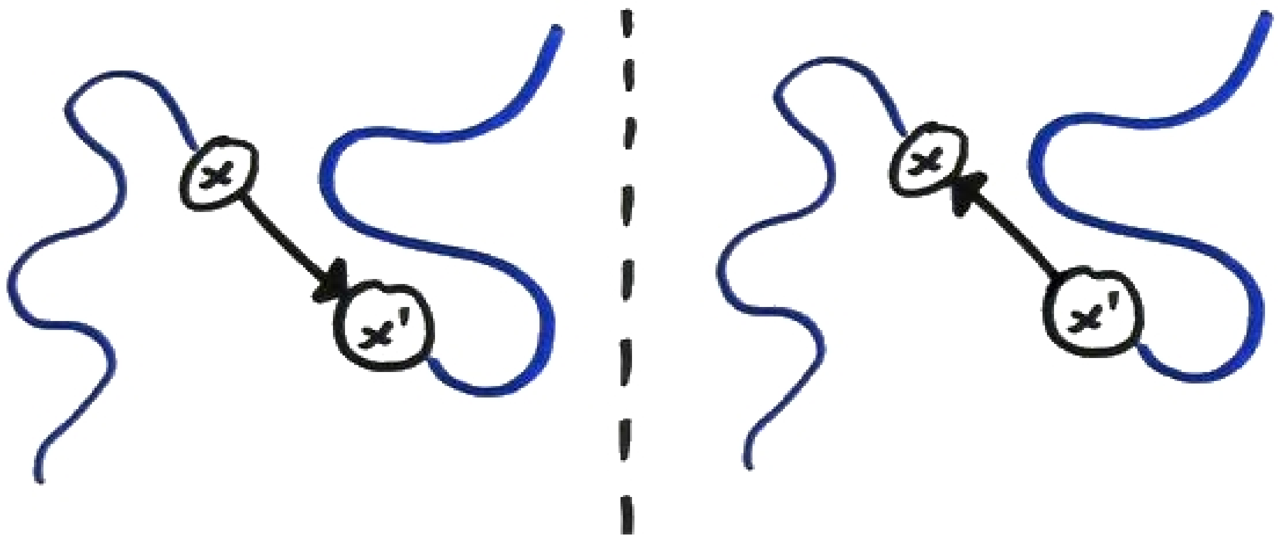
Also $P^\star$ is *the* **equilibrium distribution** of T:

To machine precision: $T^{100} \left(\begin{smallmatrix} 1 \\ 0 \\ 0 \end{smallmatrix}\right) = \left(\begin{smallmatrix} 3/5 \\ 1/5 \\ 1/5 \end{smallmatrix}\right) = P^\star$

Ergodicity requires: $T^K(x' \!\leftarrow\! x) \!>\! 0$ for all $P^\star(x') > 0$, for some K

Detailed Balance

Detailed balance means $\rightarrow x \rightarrow x'$ and $\rightarrow x' \rightarrow x$ are equally probable:



$$T(x' \leftarrow x)P^\star(x) = T(x \leftarrow x')P^\star(x')$$

Detailed balance implies the invariant condition:

$$\sum_x T(x' \leftarrow x)P^\star(x) \;=\; P^\star(x')\cancelto{1}{\sum_x T(x \leftarrow x')}$$

Enforcing detailed balance is easy: it only involves isolated pairs

Reverse operators

If T satisfies stationarity, we can define a *reverse operator*

$$\widetilde{T}(x \leftarrow x') \propto T(x' \leftarrow x)\, P^\star(x) = \frac{T(x' \leftarrow x)\, P^\star(x)}{\sum_x T(x' \leftarrow x)\, P^\star(x)} = \frac{T(x' \leftarrow x)\, P^\star(x)}{P^\star(x')}$$

Generalized balance condition:

$$T(x' \leftarrow x) P^\star(x) \;=\; \widetilde{T}(x \leftarrow x') P^\star(x')$$

also implies the invariant condition *and is necessary*.

Operators satisfying detailed balance are their own reverse operator.

Metropolis–Hastings

Transition operator

- Propose a move from the current state $Q(x'; x)$, e.g. $\mathcal{N}(x, \sigma^2)$
- Accept with probability $\min\!\left(1, \frac{P(x')Q(x;x')}{P(x)Q(x';x)}\right)$
- Otherwise next state in chain is a copy of current state

Notes

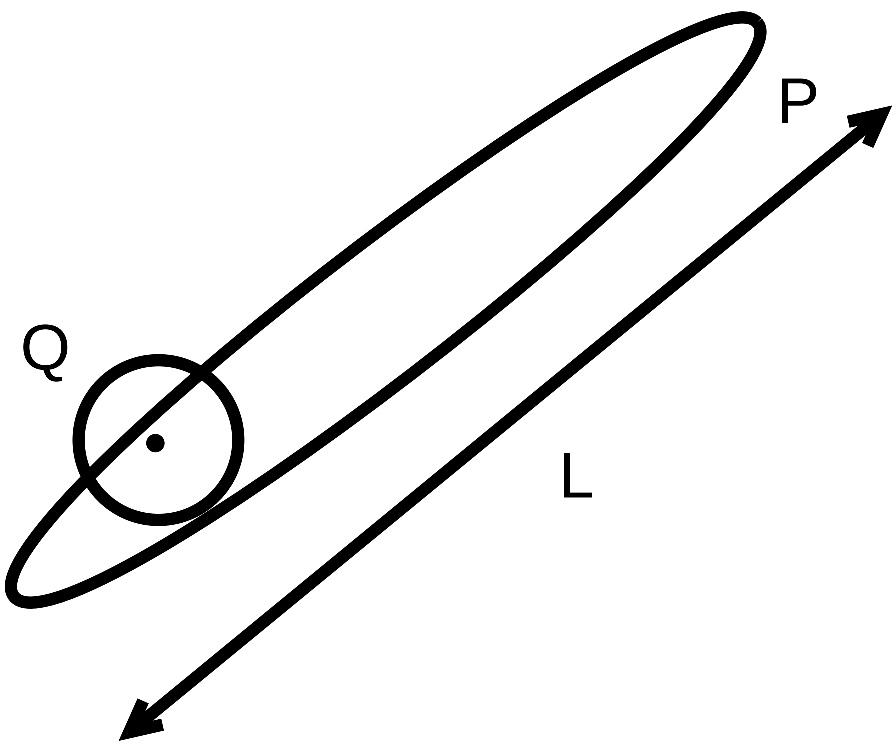
- Can use $\tilde{P} \propto P(x)$; normalizer cancels in acceptance ratio
- Satisfies detailed balance (shown below)
- Q must be chosen to fulfill the other technical requirements

---

$$P(x) \cdot T(x' \leftarrow x) = P(x) \cdot Q(x'; x) \min\left(1,\ \frac{P(x')Q(x; x')}{P(x)Q(x'; x)}\right) = \min\Big(P(x)Q(x'; x),\ P(x')Q(x; x')\Big)$$

$$= P(x') \cdot Q(x; x') \min\left(1,\ \frac{P(x)Q(x'; x)}{P(x')Q(x; x')}\right) = P(x') \cdot T(x \leftarrow x')$$

Metropolis–Hastings



Generic proposals use
$Q(x'; x) = \mathcal{N}(x, \sigma^2)$

σ **large** $\rightarrow$ **many rejections**

σ **small** $\rightarrow$ **slow diffusion:**
$\sim(L/\sigma)^2$ iterations required

Combining operators

A sequence of operators, each with $P^\star$ invariant:

$$
\begin{aligned}
&x_0 \sim P^\star(x) & & \\
&x_1 \sim T_a(x_1 \!\leftarrow\! x_0) & P(x_1) \;&=\; \textstyle\sum_{x_0} T_a(x_1 \!\leftarrow\! x_0) P^\star(x_0) \;=\; P^\star(x_1) \\
&x_2 \sim T_b(x_2 \!\leftarrow\! x_1) & P(x_2) \;&=\; \textstyle\sum_{x_1} T_b(x_2 \!\leftarrow\! x_1) P^\star(x_1) \;=\; P^\star(x_2) \\
&x_3 \sim T_c(x_3 \!\leftarrow\! x_2) & P(x_3) \;&=\; \textstyle\sum_{x_1} T_c(x_3 \!\leftarrow\! x_2) P^\star(x_2) \;=\; P^\star(x_3) \\
&\quad\cdots & &\cdots
\end{aligned}
$$

— Combination $T_c T_b T_a$ leaves $P^\star$ invariant

— If they can reach any x, $T_c T_b T_a$ is a valid MCMC operator

— Individually T_c, T_b and T_a need not be ergodic

Gibbs sampling

A method with no rejections:

– Initialize $\mathbf{z}$ to some value
– Pick each variable in turn or randomly and resample $P(z_i|\mathbf{z}_{j\neq i})$

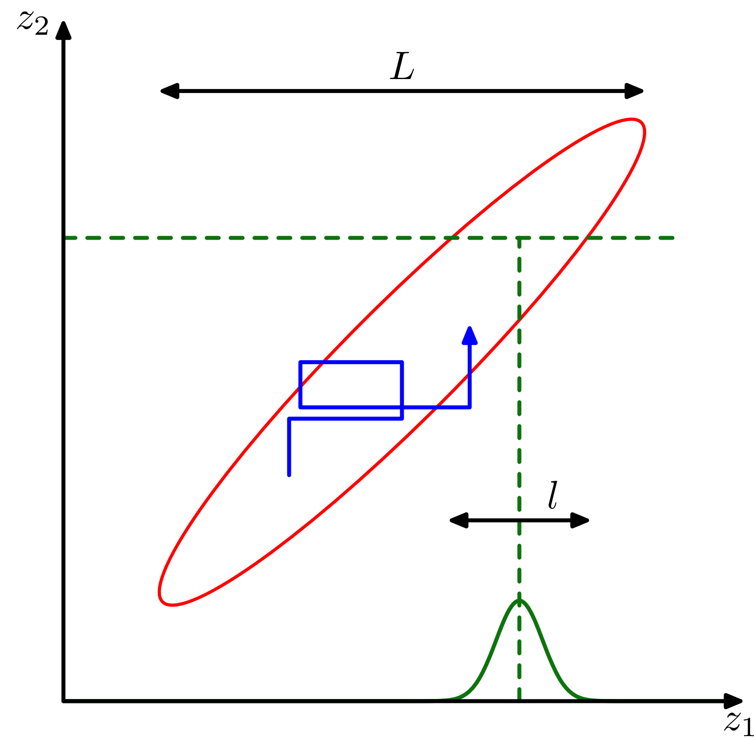


Figure from PRML, Bishop (2006)

Proof of validity:

Metropolis–Hastings 'proposals' $P(z_i|\mathbf{z}_{j\neq i}) \Rightarrow$ accept with prob. 1
Apply a series of these operators; don't need to check acceptance

Gibbs sampling

Alternative explanation:

Chain is currently at $\mathbf{x}$

At equilibrium can assume $\mathbf{x} \sim P(\mathbf{x})$

Consistent with $\mathbf{x}_{j \neq i} \sim P(\mathbf{x}_{j \neq i}), \quad x_i \sim P(x_i \,|\, \mathbf{x}_{j \neq i})$

Pretend x_i was never sampled and do it again.

This view may be useful later for non-parametric applications

“Routine” Gibbs sampling

Gibbs sampling benefits from few free choices and convenient features of conditional distributions:

- Conditionals with a few discrete settings can be explicitly normalized:

$$P(x_i|\mathbf{x}_{j\neq i}) \;\propto\; P(x_i, \mathbf{x}_{j\neq i})$$
$$= \frac{P(x_i, \mathbf{x}_{j\neq i})}{\sum_{x'_i} P(x'_i, \mathbf{x}_{j\neq i})} \;\leftarrow \text{this sum is small and easy}$$

- Continuous conditionals only univariate
  $\Rightarrow$ amenable to standard sampling methods.

`WinBUGS` and `OpenBUGS` sample graphical models using these tricks

Auxiliary variables

The point of MCMC is to marginalize out variables,
but one can introduce more variables:

$$\int f(x)P(x)\,\mathrm{d}x = \int f(x)P(x,v)\,\mathrm{d}x\,\mathrm{d}v$$

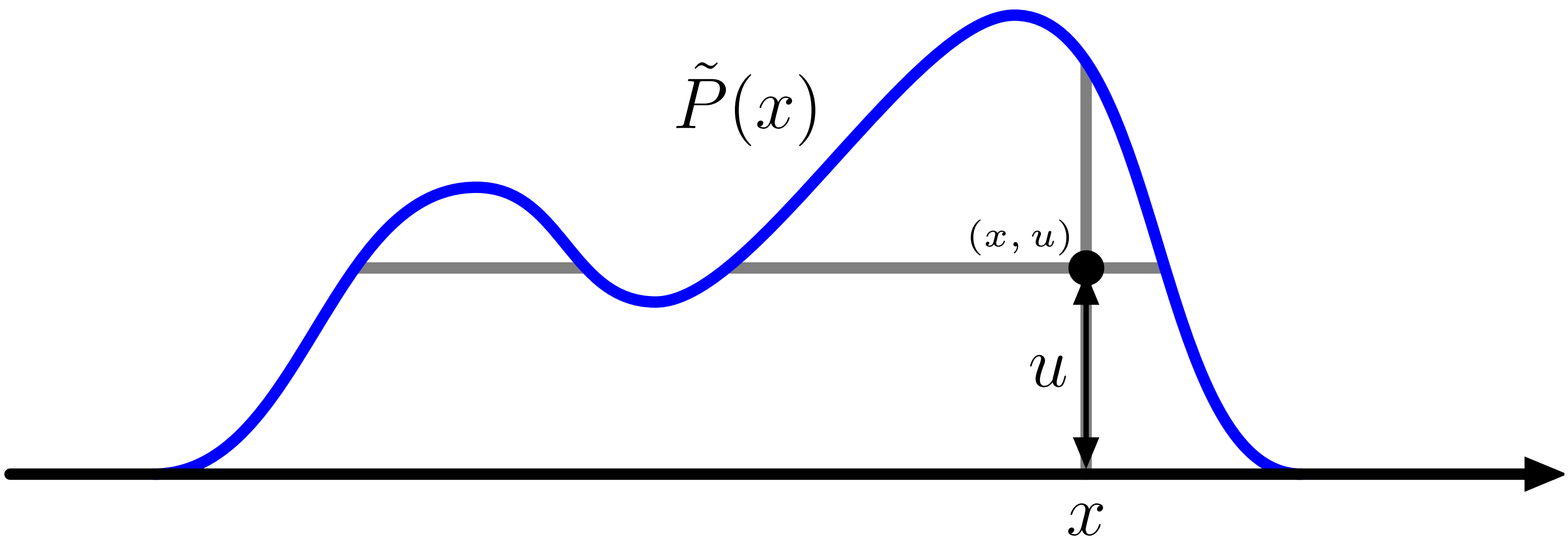
$$\approx \frac{1}{S}\sum_{s=1}^{S} f(x^{(s)}), \quad x,v \sim P(x,v)$$

We might want to do this if

- $P(x|v)$ and $P(v|x)$ are simple
- $P(x,v)$ is otherwise easier to navigate

Slice sampling idea

Sample point uniformly under curve $\tilde{P}(x) \propto P(x)$

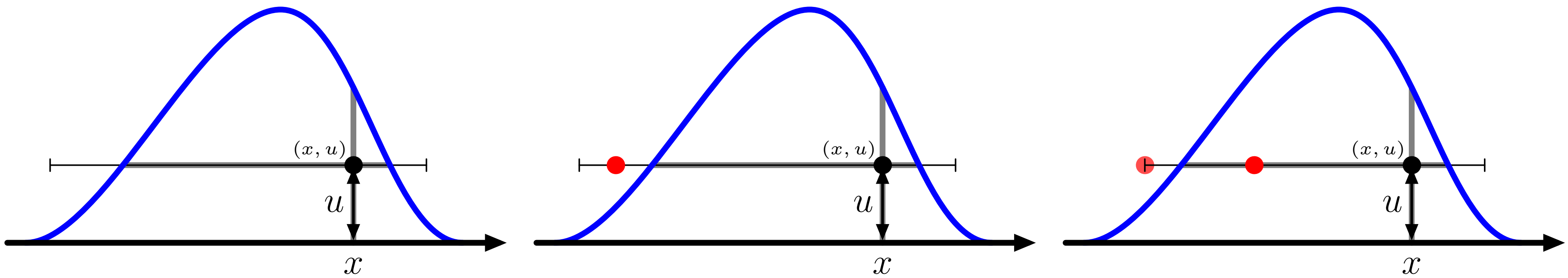


$$p(u|x) = \text{Uniform}[0, \tilde{P}(x)]$$

$$p(x|u) \propto \begin{cases} 1 & \tilde{P}(x) \geq u \\ 0 & \text{otherwise} \end{cases} = \text{“Uniform on the slice”}$$

Slice sampling

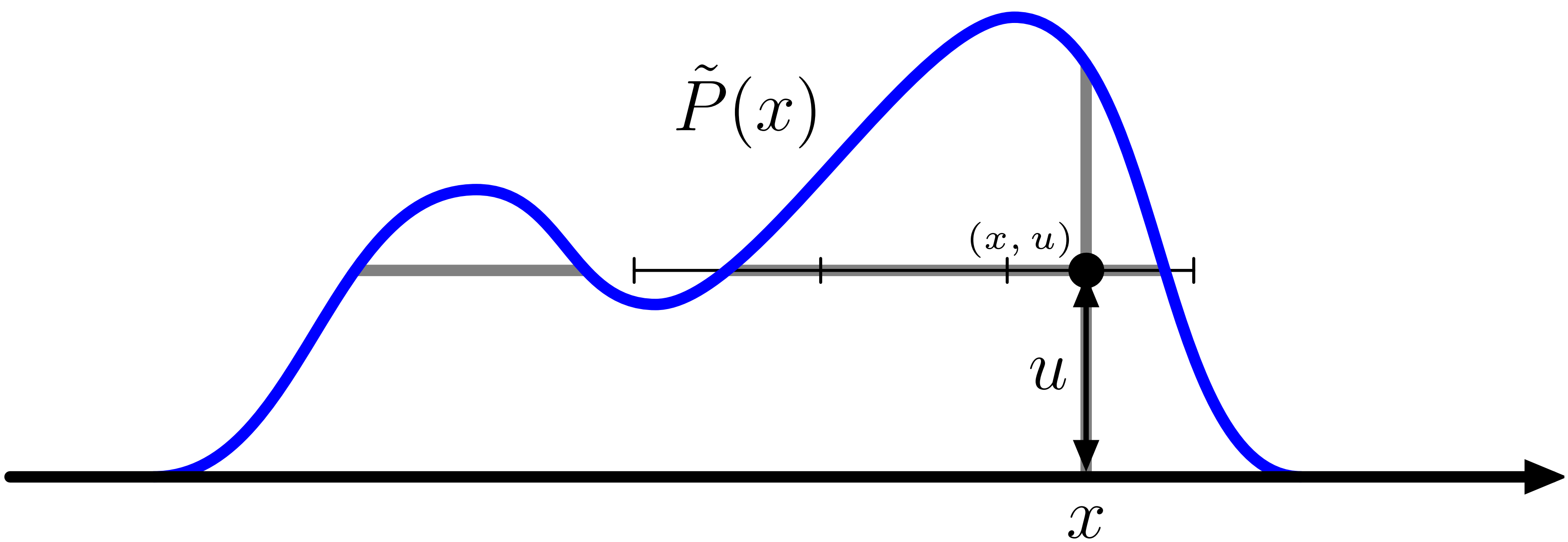
Unimodal conditionals



- bracket slice
- sample uniformly within bracket
- shrink bracket if $\tilde{P}(x) < u$ (off slice)
- accept first point on the slice

Slice sampling

Multimodal conditionals



- place bracket randomly around point
- linearly step out until bracket ends are off slice
- sample on bracket, shrinking as before

Satisfies detailed balance, leaves $p(x|u)$ invariant

Slice sampling

Advantages of slice-sampling:

- Easy — only require $\tilde{P}(x) \propto P(x)$ pointwise
- No rejections
- Step-size parameters less important than Metropolis

[More advanced versions of slice sampling have been developed]

Hamiltonian dynamics

Construct a landscape with gravitational potential energy, E(x):

$$P(x) \propto e^{-E(x)}, \qquad E(x) = -\log P^*(x)$$

Introduce velocity v carrying kinetic energy $K(v) = v^\top v/2$

Some physics:

- Total energy or Hamiltonian, $H = E(x) + K(v)$
- Frictionless ball rolling $(x, v) \rightarrow (x', v')$ satisfies $H(x', v') = H(x, v)$
- Ideal Hamiltonian dynamics are time reversible:
  - reverse v and the ball will return to its start point

Hamiltonian Monte Carlo

Define a joint distribution:

- $P(x, v) \propto e^{-E(x)} e^{-K(v)} = e^{-E(x)-K(v)} = e^{-H(x,v)}$
- Velocity independent of position and Gaussian distributed

Markov chain operators

- Gibbs sample velocity
- Simulate Hamiltonian dynamics then flip sign of velocity
  - Hamiltonian 'proposal' is deterministic and reversible $q(x', v'; x, v) = q(x, v; x', v') = 1$
  - Conservation of energy means $P(x, v) = P(x', v')$
  - Metropolis acceptance probability is 1

Except we can't simulate Hamiltonian dynamics exactly

Leap-frog dynamics

a discrete approximation to Hamiltonian dynamics:

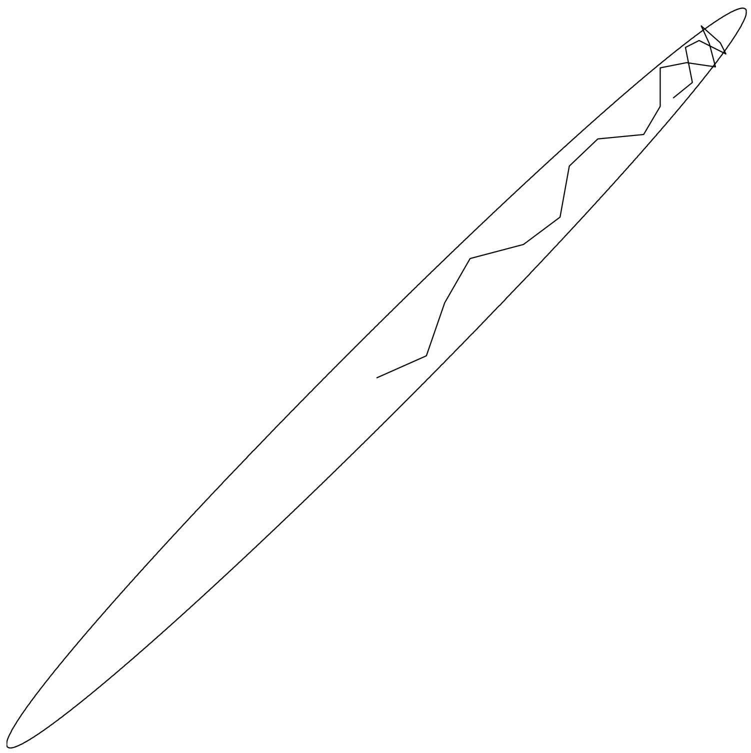
$$\begin{aligned}
v_i(t + \tfrac{\epsilon}{2}) &= v_i(t) - \frac{\epsilon}{2} \frac{\partial E(x(t))}{\partial x_i} \\
x_i(t + \epsilon) &= x_i(t) + \epsilon v_i(t + \tfrac{\epsilon}{2}) \\
p_i(t + \epsilon) &= v_i(t + \tfrac{\epsilon}{2}) - \frac{\epsilon}{2} \frac{\partial E(x(t+\epsilon))}{\partial x_i}
\end{aligned}$$

- H is not conserved
- dynamics are still deterministic and reversible
- Acceptance probability becomes $\min[1, \exp(H(v,x) - H(v',x'))]$

Hamiltonian Monte Carlo

The algorithm:

- Gibbs sample velocity $\sim \mathcal{N}(0, 1)$
- Simulate Leapfrog dynamics for L steps
- Accept new position with probability $\min[1, \exp(H(v, x) - H(v', x'))]$

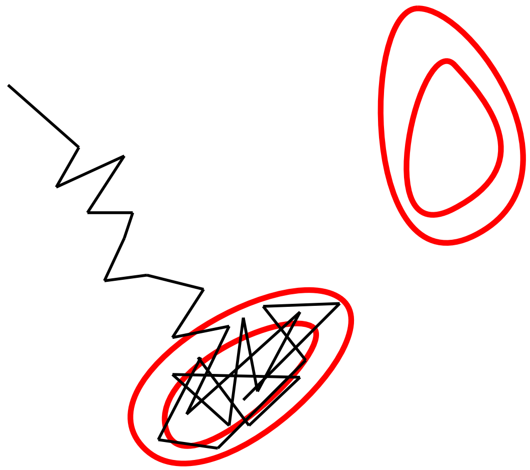


The original name is **Hybrid Monte Carlo**, with reference to the dynamical simulation method on which it was based.

MCMC's main problem

Mixing:

Efficient burn-in and mode exploration



Sampling summary

- Probabilistic modelling requires the computation of many sums and integrals
- Sampling looks noisy and inefficient,
  but is highly competitive on the most complex problems
- Monte Carlo does not explicitly depend on dimension,
  although the global methods work only in low dimensions
- Markov chain Monte Carlo (MCMC) uses simple,
  local computations $\Rightarrow$ "easy" to implement
  (harder to diagnose).

Methods:

– Direct, rejection and importance sampling
– MCMC: Metropolis–Hastings, Gibbs and Slice sampling,. . .