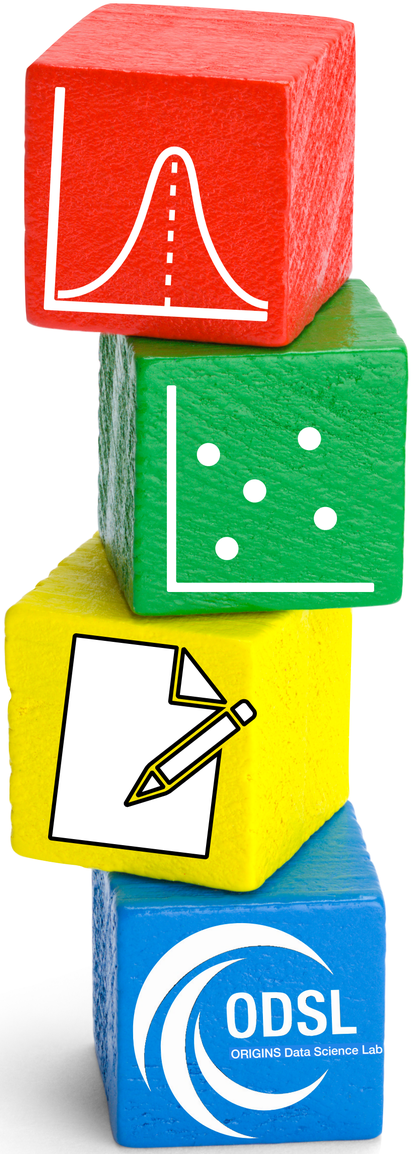
ODSL Statistics Block Course

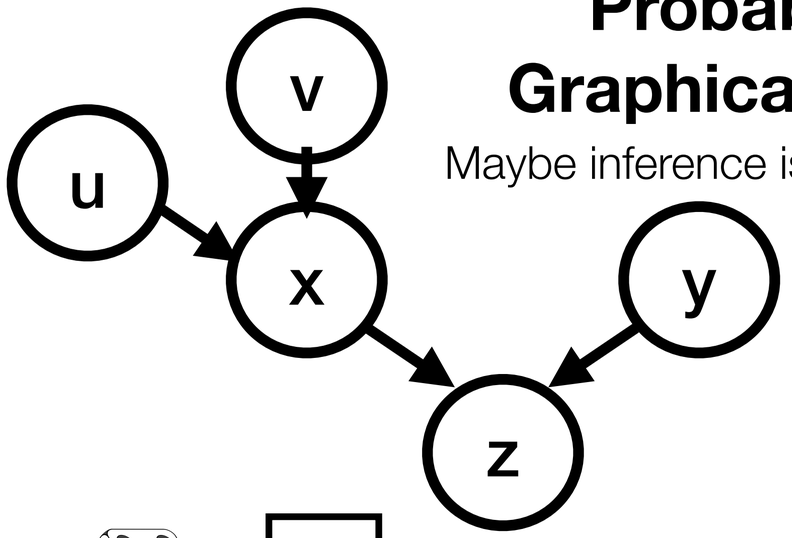
Lecture 5: Learning to sample
A.k.a, what to do when inference is intractable

Nicole Hartman
nicole.hartman@tum.de

24 Sept 2025



Summary



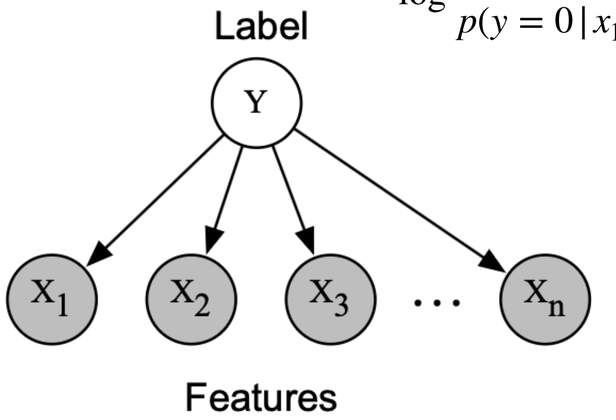
Probabilistic Graphical models

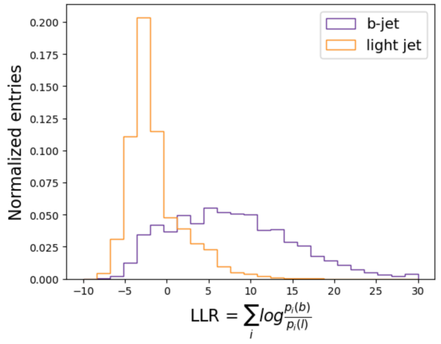
Maybe inference is no longer analytic.

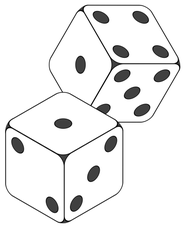
Simplifying assumptions

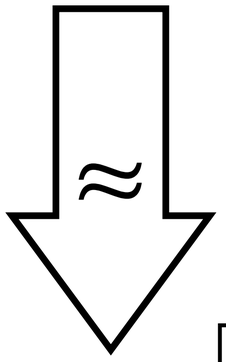
Naive Bayes

$$\log \frac{p(y=1\,|\,x_1,\ldots x_n)}{p(y=0\,|\,x_1,\ldots x_n)} = \sum_i \log \frac{p(x_i\,|\,y=1)}{p(x_i\,|\,y=0)}$$





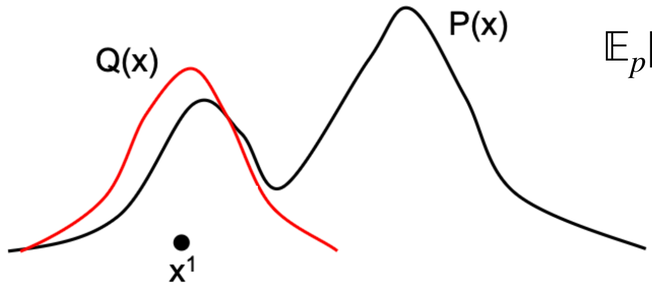




Monte Carlo sampling

$$\mathbb{E}_p[f(x)] \approx \sum_{x_i \sim p} f(x)$$

importance sampling



$$\mathbb{E}_p[f(x)] \approx \sum_{x_i \sim Q} f(x)\frac{P(x)}{Q(x)}$$

importance weight, $w(x)$

Your turn !

$p(d, i, g, s, l) =$

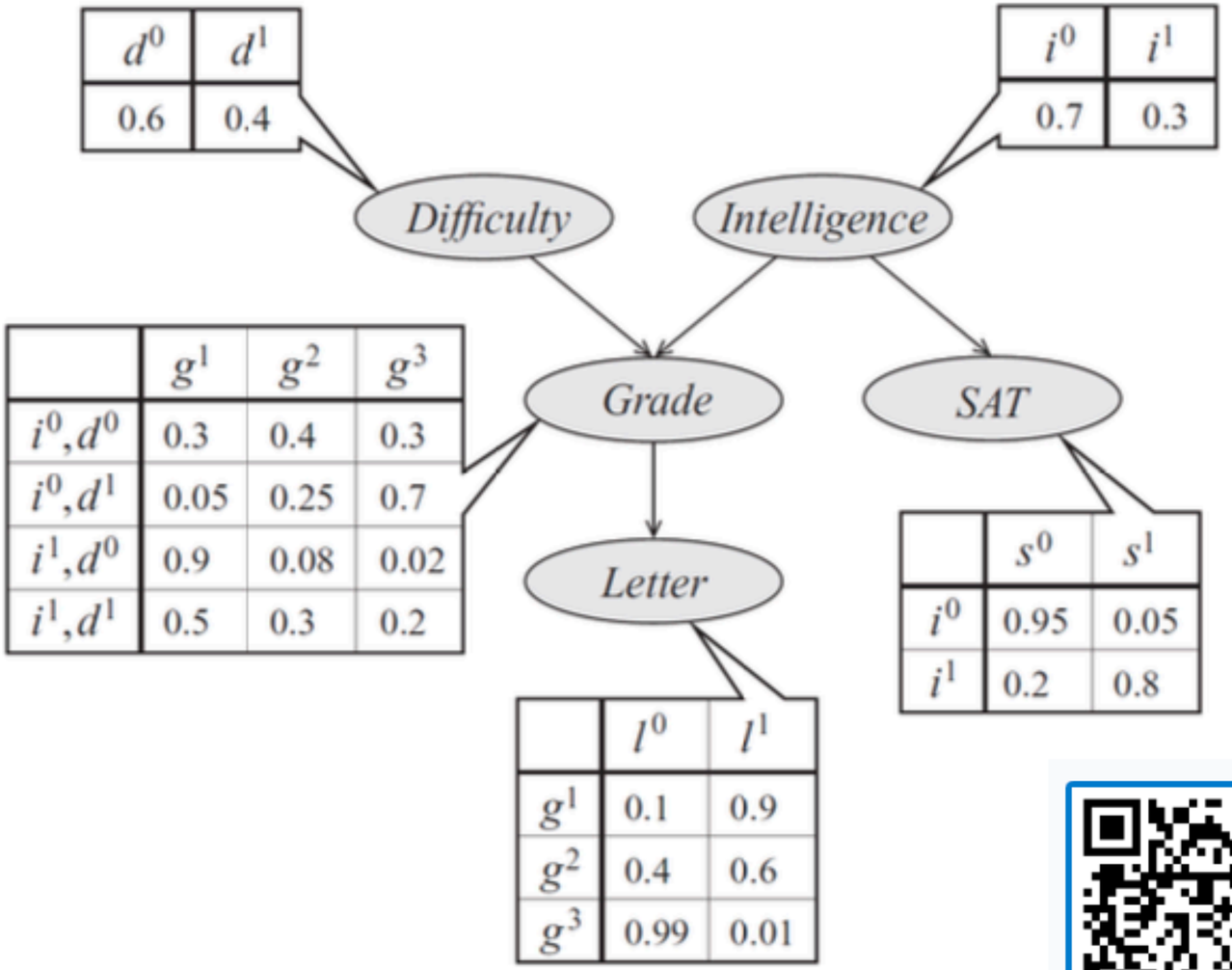
A) $p(l \mid d, i, g, s)\ p(d \mid i, g, s)\ p(i \mid g, s)\ p(g \mid s)\ p(s)$

B) $p(d \mid i, g, s, l)\ p(i \mid g, s, l)\ p(g \mid s, l)\ p(s \mid l)\ p(l)$

C) $p(l \mid g, s)\ p(s \mid i, g)$

D) $p(l \mid g)\ p(s \mid i) p(g \mid d, i)$

E) $p(l \mid g)\ p(s \mid i) p(g \mid d, i) p(d) p(i)$



d^0	d^1
0.6	0.4

i^0	i^1
0.7	0.3

	g^1	g^2	g^3
i^0, d^0	0.3	0.4	0.3
i^0, d^1	0.05	0.25	0.7
i^1, d^0	0.9	0.08	0.02
i^1, d^1	0.5	0.3	0.2

	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

	l^0	l^1
g^1	0.1	0.9
g^2	0.4	0.6
g^3	0.99	0.01

PollEv.com/nicolehartman968

Agenda

- Markov Chain Monte Carlo
  - Example: Metropolis Hastings
  - Formalism
  - Practical tips
- Examples from in science

Gratefully borrowing the presentation from Stefano Ermon's CS228 course

(He creditted Eric Xing, Qirong Ho (CMU) and David Sontag (NYU), so I link Eric's and David's lectures below as well:

https://www.youtube.com/watch?v=oCRqPe-MlYc

https://people.csail.mit.edu/dsontag/courses/pgm13/slides/lecture9.pdf

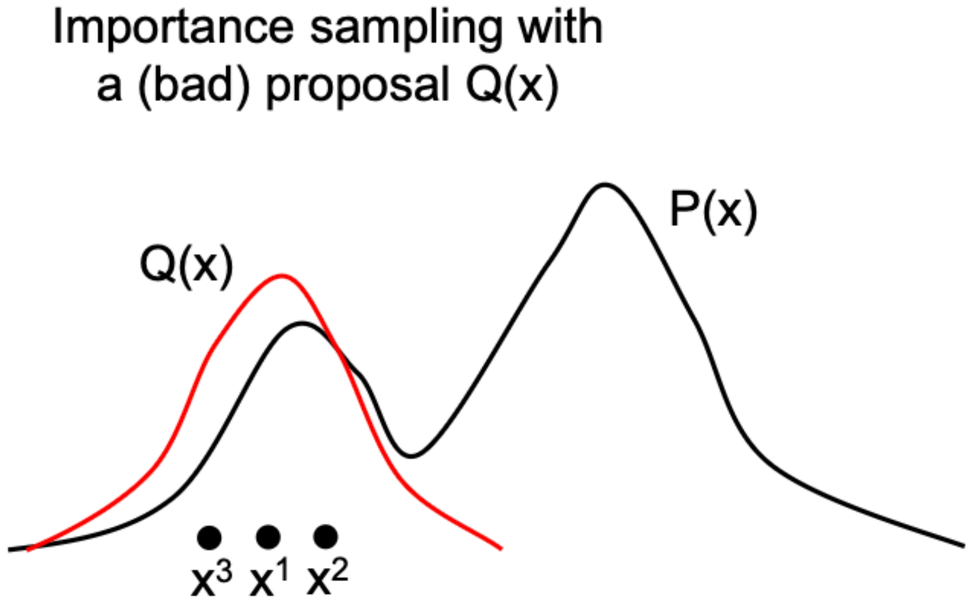
Unless otherwise stated, I've used their slides in the following 🌞

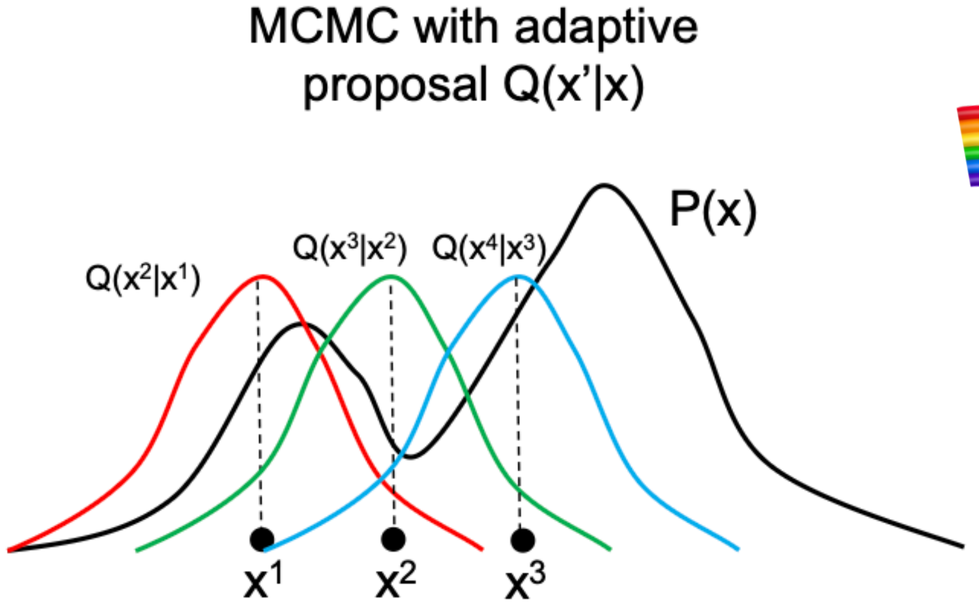
Markov Chain Monte Carlo (MCMC)

MCMC algorithms feature adaptive proposals

↪ Instead of Q(x'), use Q(x' | x)

- x': new state being sampled
- x: previous sample





MCMC example: Metropolis Hastings

1. Sample $x' \sim Q(x'|x)$, where x = prev sample

$$A(x'|x) = \min\left(1, \frac{P(x')}{P(x)}\frac{Q(x|x')}{Q(x'|x)}\right)$$

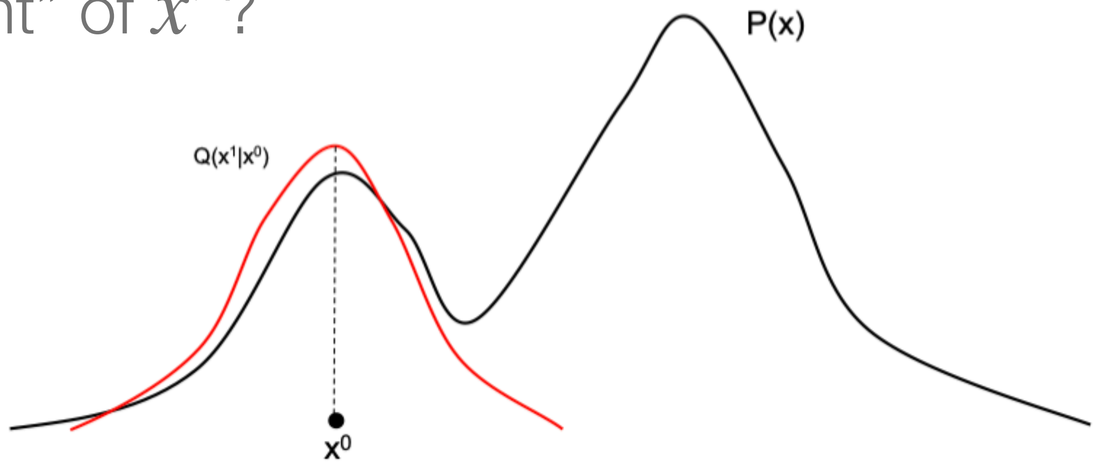
Q: What would be the "importance weight" of x' ?

A) $Q(x'|x)$

B) $P(x')$

C) $P(x')/Q(x'|x)$

D) $P(x)/Q(x|x')$



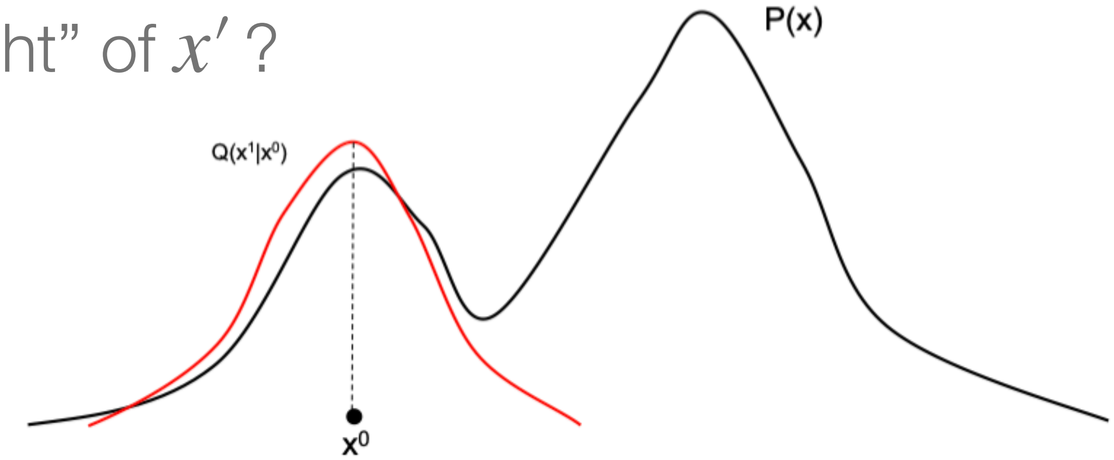
MCMC example: Metropolis Hastings

1. Sample $x' \sim Q(x' \mid x)$, where x = prev sample

$$A(x' \mid x) = \min\left(1, \frac{P(x')}{P(x)} \frac{Q(x \mid x')}{Q(x' \mid x)}\right)$$

Q: What would be the "importance weight" of x' ?

A:



MCMC example: Metropolis Hastings

1. Sample $x' \sim Q(x'|x)$, where x = prev sample

2. The new sample x' is accepted or rejected with probability $A(x'|x)$

$$A(x'|x) = \min\left(1, \frac{P(x')}{P(x)}\frac{Q(x|x')}{Q(x'|x)}\right)$$

Similarly, if we considered the reverse process $x \sim Q(x|x')$ …

$P(x')/Q(x'|x)$: importance weight for x'

$P(x)/Q(x|x')$: importance weight for x

$A(x'|x)$ is the ratio of importance weights!

MCMC example: Metropolis Hastings

1. Sample $x' \sim Q(x'|x)$, where x = prev sample

2. The new sample x' is accepted or rejected with probability, $A(x'|x)$

$$A(x'|x) = \min\left(1, \frac{P(x')}{P(x)}\frac{Q(x|x')}{Q(x'|x)}\right)$$

We don't need to know the probability, only the ratio: $P(x')/P(x)$.

$A(x'|x)$ ensures that (after sufficiently many draws) the samples will come from the the true distribution $P(x)$.

[Proof later in lecture]

MCMC example: Metropolis Hastings

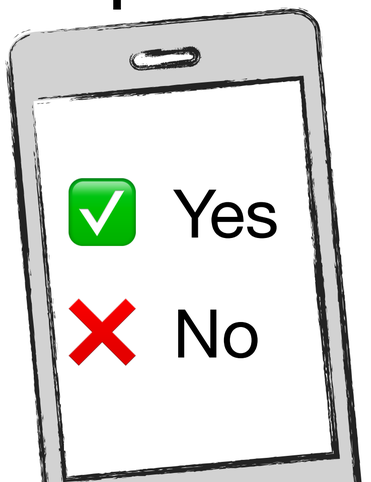
1. Sample $x' \sim Q(x'|x)$, where x = prev sample

2. The new sample x' is accepted or rejected with probability, $A(x'|x)$

PollEv.com/nicolehartman968

$$A(x'|x) = \min\left(1, \frac{P(x')\ Q(x|x')}{P(x)\ Q(x'|x)}\right)$$

Concept check: If I want to use Metropolis-Hastings to sample from posterior $p(\theta|y) = \frac{p(x|\theta)p(\theta)}{p(y)}$, do I need to know the evidence?



MCMC example: Metropolis Hastings

1. Sample $x' \sim Q(x'|x)$, where x = prev sample

2. The new sample x' is accepted or rejected with probability, $A(x'|x)$

$$A(x'|x) = \min\left(1, \frac{P(x')}{P(x)}\frac{Q(x|x')}{Q(x'|x)}\right)$$

Concept check: If I want to use Metropolis-Hastings to sample from posterior $p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$, do I need to know the evidence?

✅ Yes

❌ No

$$\frac{p(\theta'|y)}{p(\theta|y)} = \frac{p(y|\theta')p(\theta') \;/\; p(y)}{p(y|\theta)p(\theta) \;/\; p(y)} = \frac{p(y|\theta')p(\theta')}{p(y|\theta)p(\theta)}$$

Metropolis Hastings algorithm

1. Initialize starting state $x^{(0)}$, set $t=0$
2. Burn-in: while samples have "not converged"
   - $x=x^{(t)}$
   - $t=t+1$,
   - sample $x^* \sim Q(x^*|x)$ // draw from proposal
   - sample $u \sim$ Uniform(0,1) // draw acceptance threshold
     - if $u < A(x^* \mid x) = \min\left(1, \frac{P(x^*)Q(x \mid x^*)}{P(x)Q(x^* \mid x)}\right)$
       - $x^{(t)} = x^*$ // transition
     - else
       - $x^{(t)} = x$ // stay in current state

   Function
   Draw sample (x(t))

- Take samples from P(x): Reset t=0, for $t=1{:}N$
  - x(t+1) ← Draw sample (x(t))
- (Monte Carlo Estimation using these N final samples)

The MH Algorithm

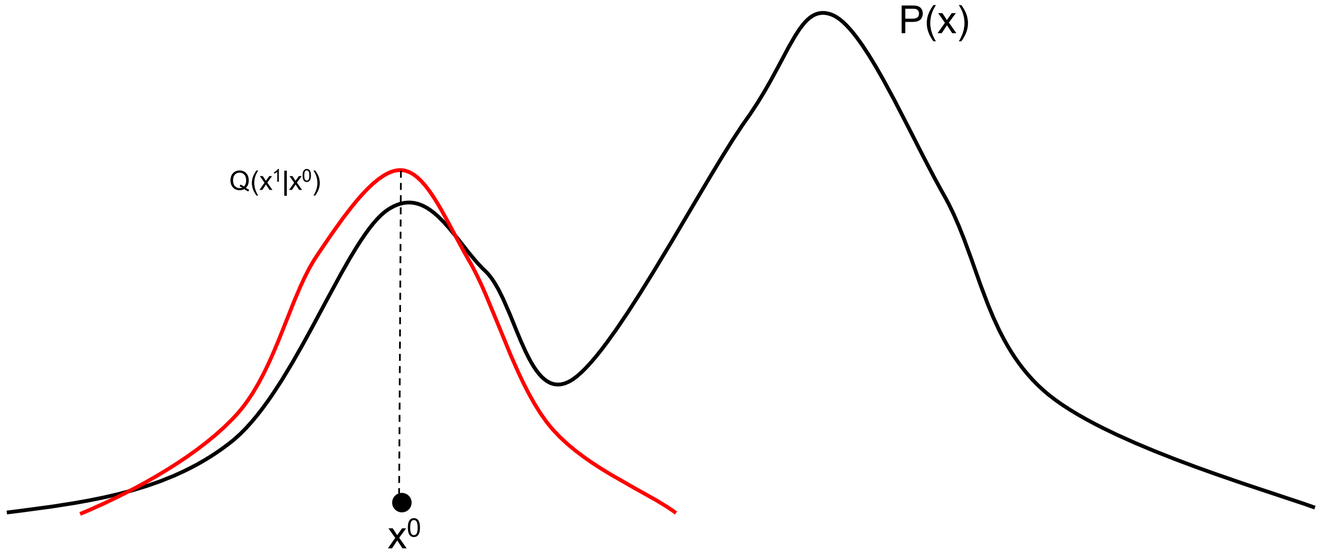
$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

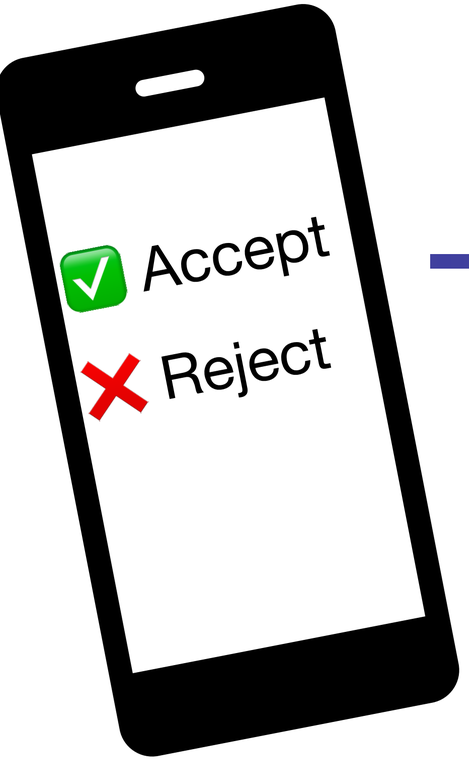
- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

PollEv.com /nicolehartman968

Initialize $x^{(0)}$

…





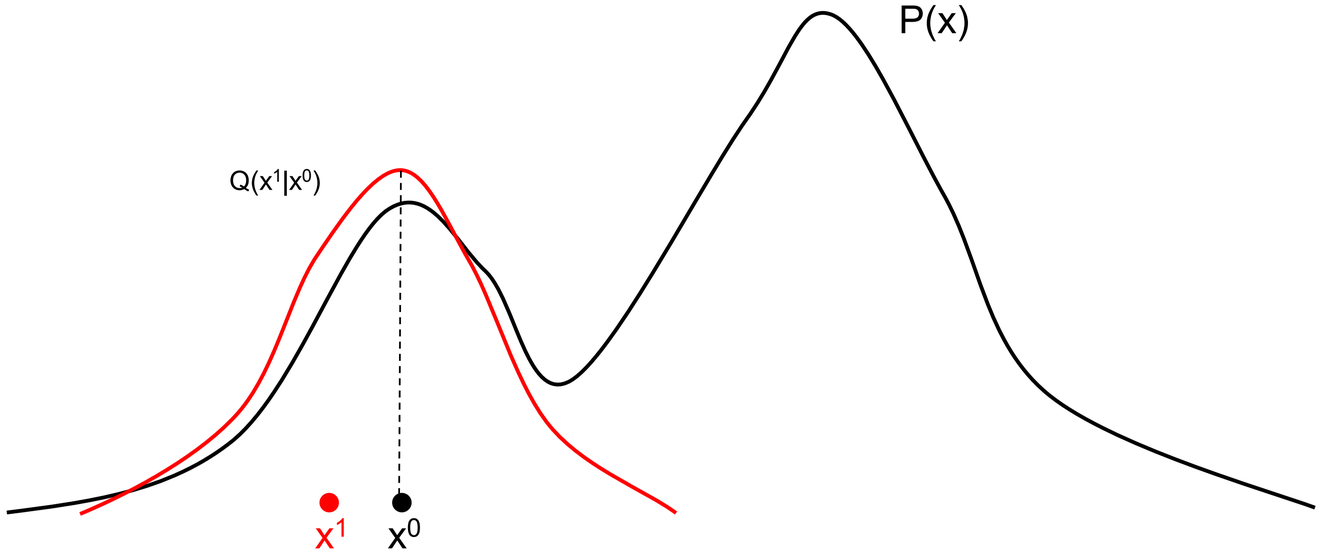
The MH Algorithm

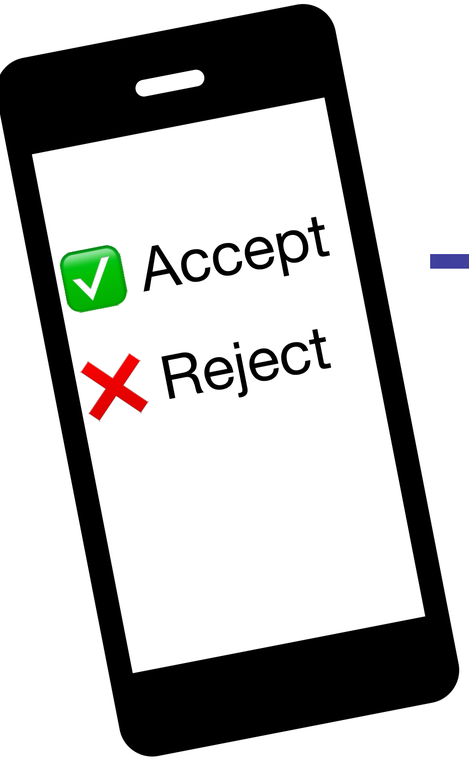
$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

PollEv.com /nicolehartman968

Initialize $x^{(0)}$
Draw,





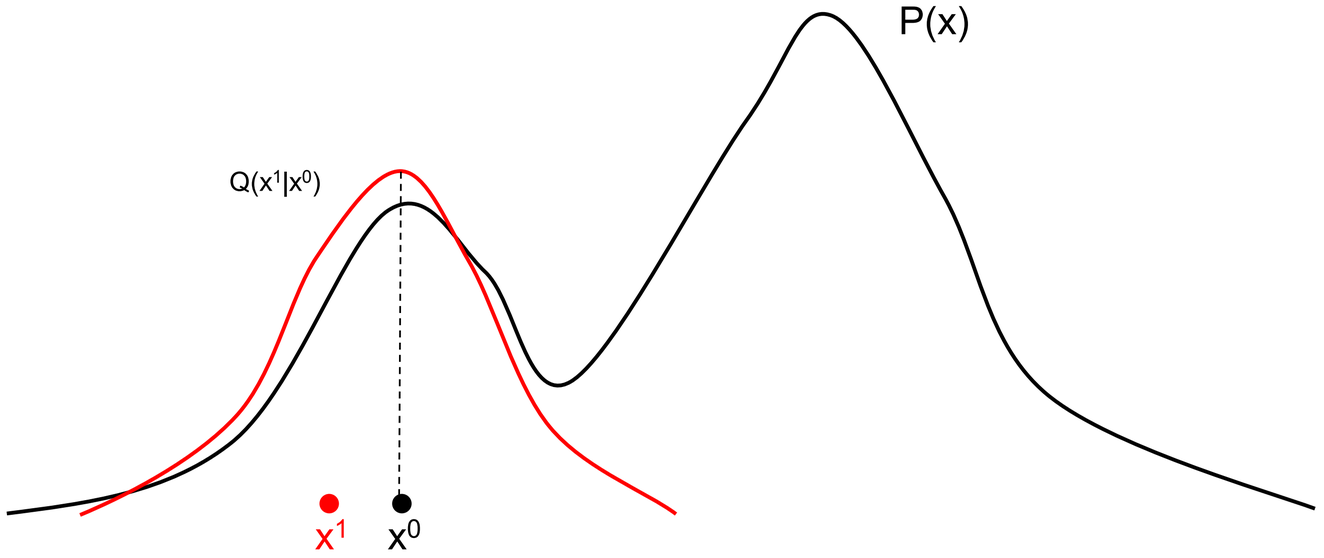
The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

PollEv.com /nicolehartman968

Initialize $x^{(0)}$
Draw, accept x^1

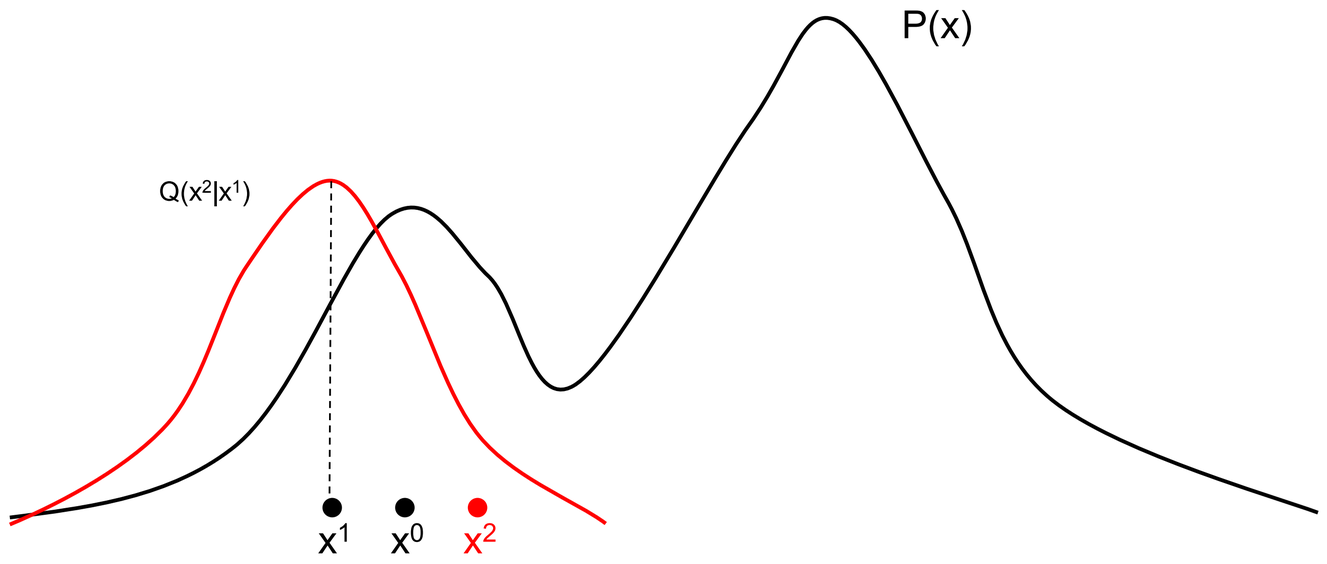


The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

Initialize $x^{(0)}$
Draw, accept x^1
Draw, accept x^2

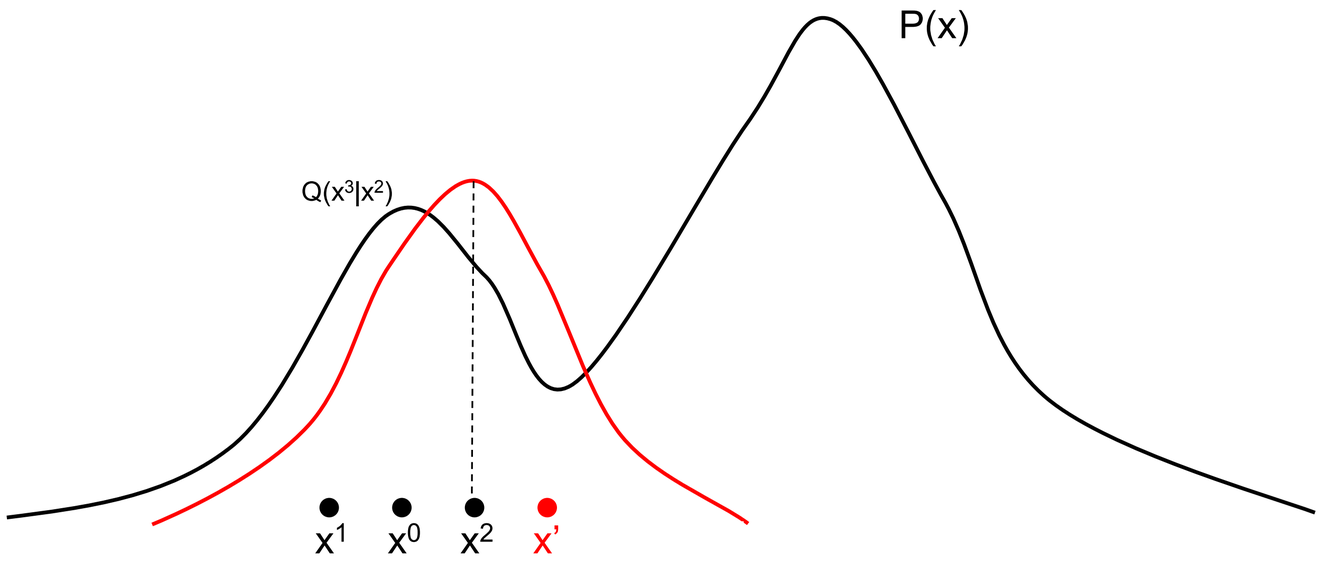


The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

Initialize $x^{(0)}$
Draw
Draw
Draw



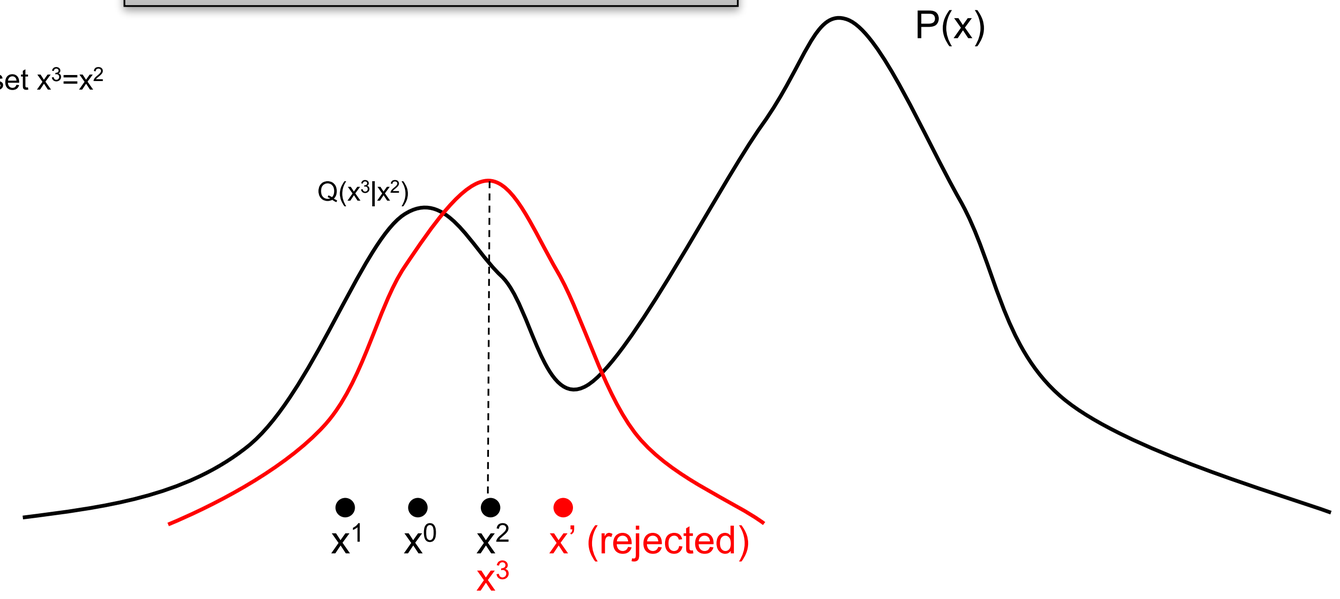
The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

Initialize $x^{(0)}$
Draw, accept x^1
Draw, accept x^2
Draw but reject; set $x^3=x^2$

We reject because $P(x')/P(x^2)$ is very small, hence $A(x'|x^2)$ is close to zero!

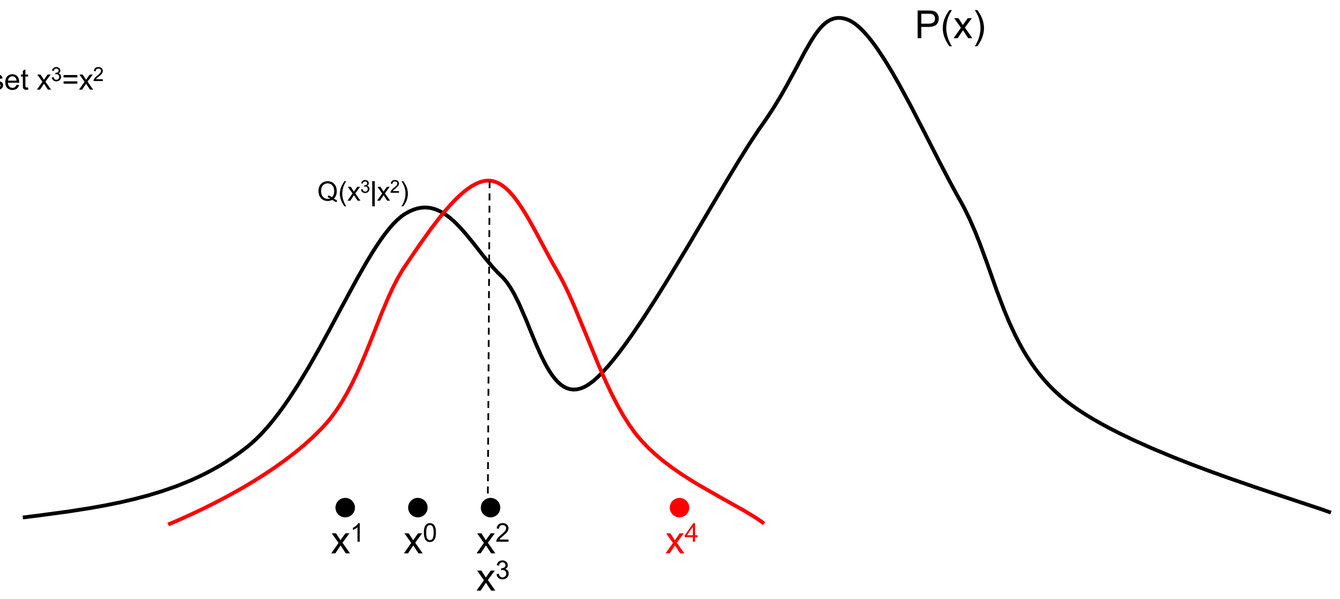


The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

Initialize $x^{(0)}$
Draw, accept x^1
Draw, accept x^2
Draw but reject; set $x^3=x^2$
Draw, accept x^4

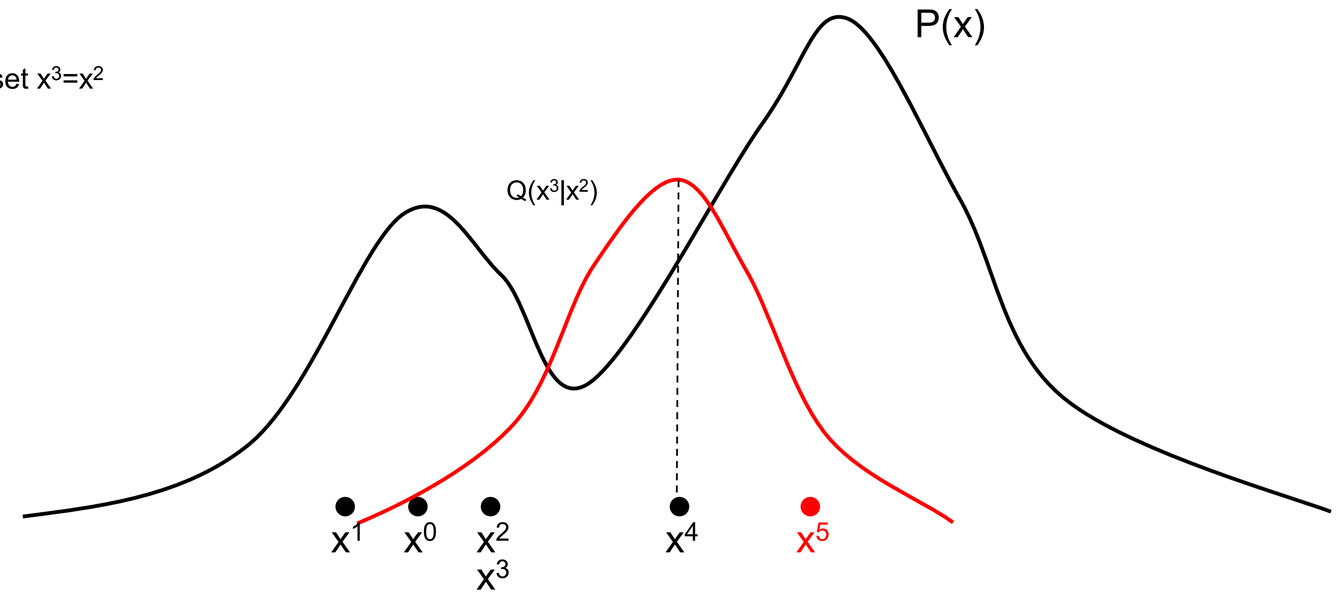


The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

Initialize $x^{(0)}$
Draw, accept x^1
Draw, accept x^2
Draw but reject; set $x^3=x^2$
Draw, accept x^4
Draw, accept x^5



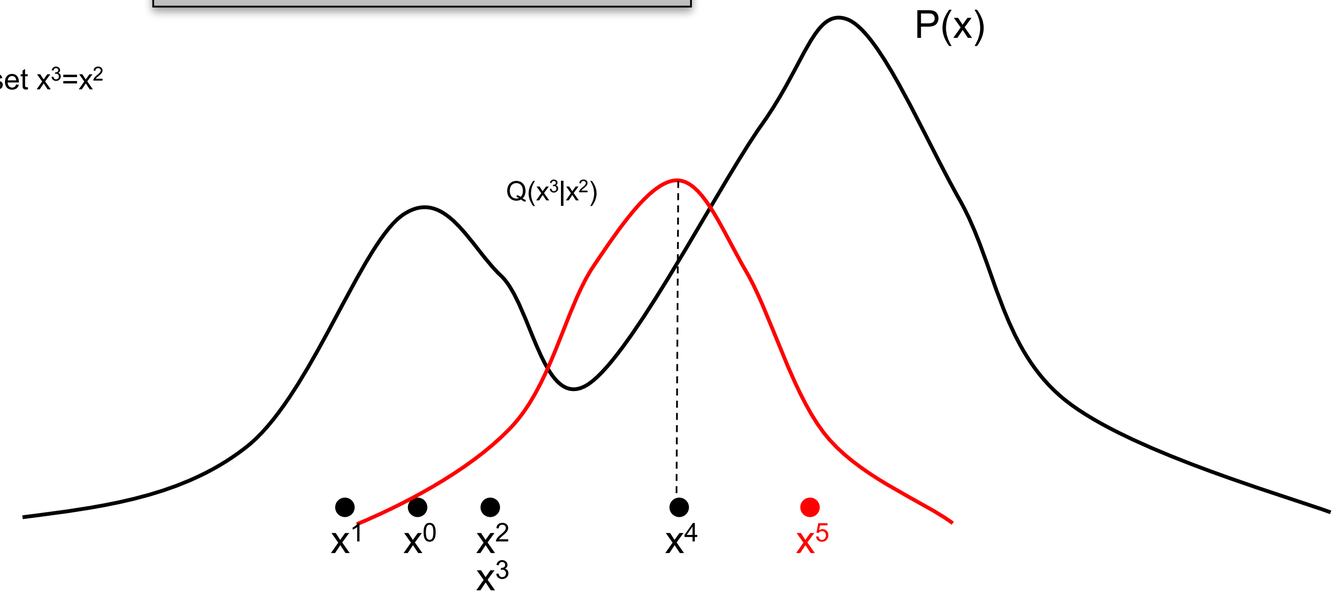
The MH Algorithm

$$A(x'|x) = \min\left(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}\right)$$

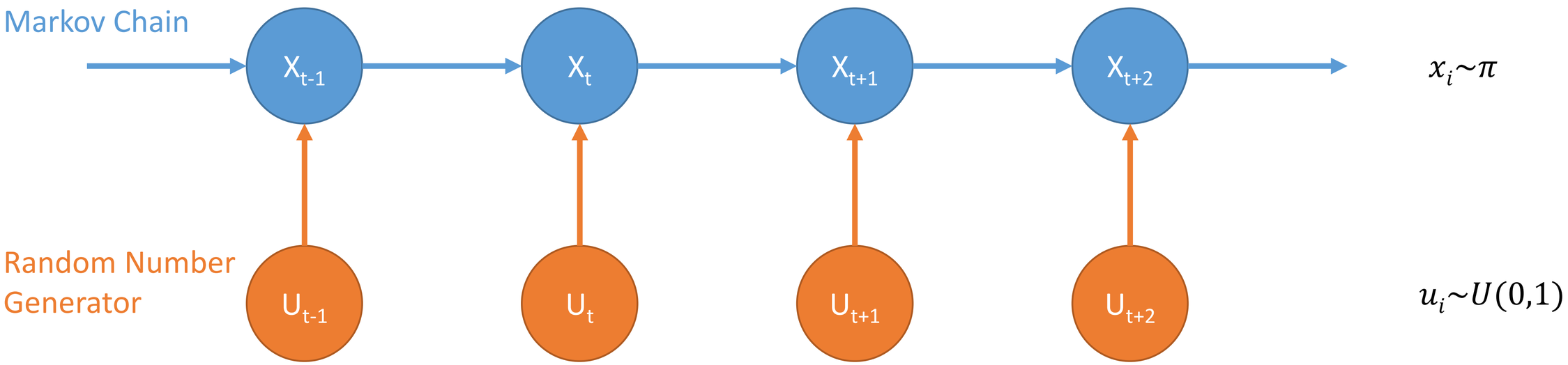
- Example:
  - Let Q(x'|x) be a Gaussian centered on x
  - We're trying to sample from a bimodal distribution P(x)

Initialize $x^{(0)}$
Draw, accept x^1
Draw, accept x^2
Draw but reject; set $x^3=x^2$
Draw, accept x^4
Draw, accept x^5

The adaptive proposal Q(x'|x) allows us to sample both modes of P(x)!

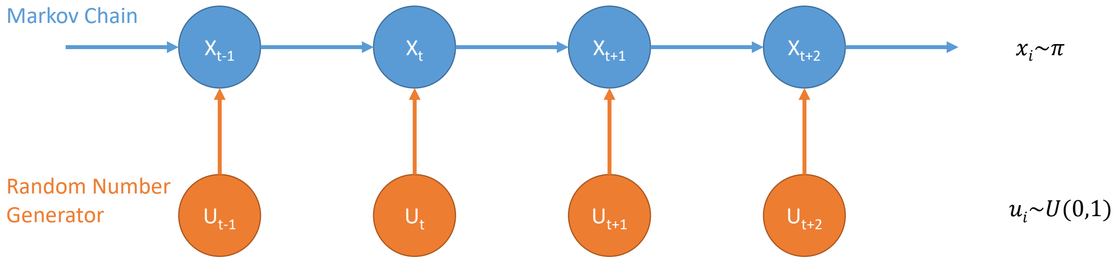


Markov Chain



Markov Chain

Defn: Markovian

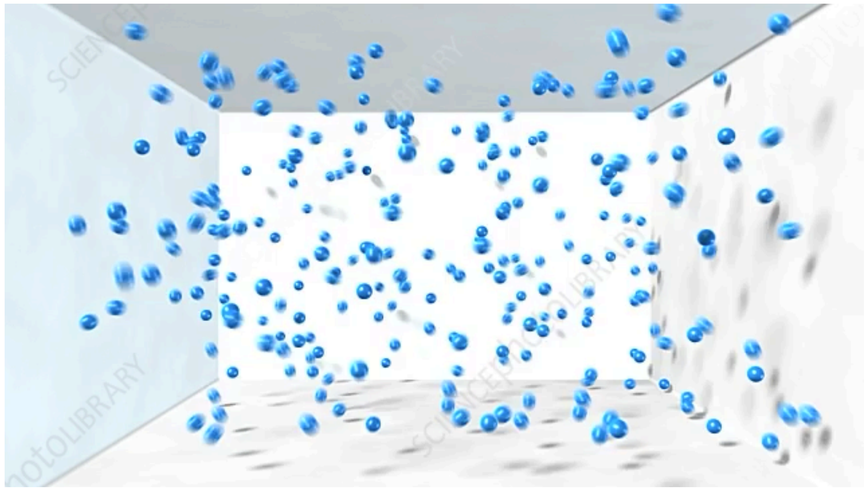


The probability distribution of a state depends only on the current state and not on the past states.

$$P(x_t \mid x_{t-1}, x_{t-2} \ldots, x_1) = P(x_t \mid x_{t-1})$$

Ex: Random walk

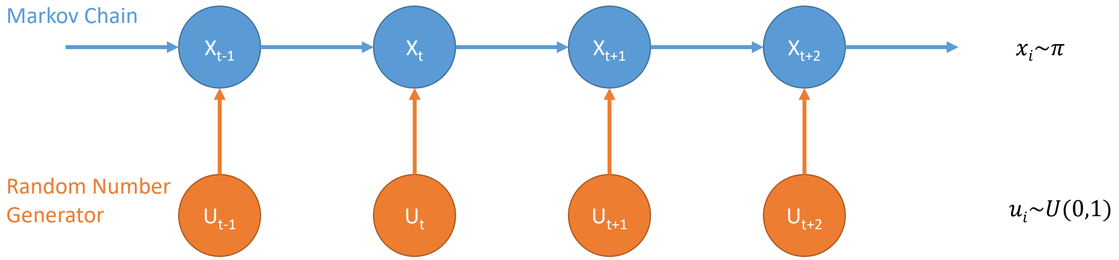
$$x_t = (\vec{x}_t, \vec{v}_t)$$



Markov Chain

Defn: Markovian

$$P(x_t \mid x_{t-1}, x_{t-2} \ldots, x_1) = P(x_t \mid x_{t-1})$$



Concept check: What if I wanted to find the probability distribution over all the states in the history, $P(x_t, x_{t-1}, \ldots, x_2, x_1)$?

A) $P(x_t \mid x_{t-1}, \ldots x_2, x_1) \ldots P(x_2 \mid x_1) p(x_1)$

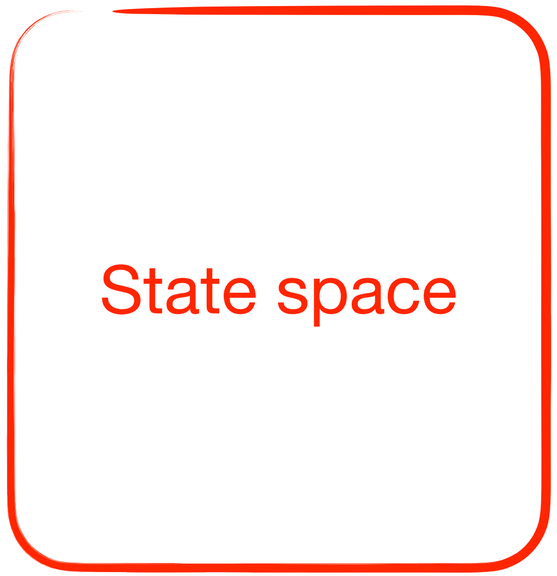
B) $P(x_t \mid x_{t-1})$

C) $P(x_t \mid x_{t-1}) P(x_{t-1} \mid x_{t-2}) \ldots P(x_2 \mid x_1) P(x_1)$

D) $P(x_t \mid x_{t-1})$

PollEv.com /nicolehartman968

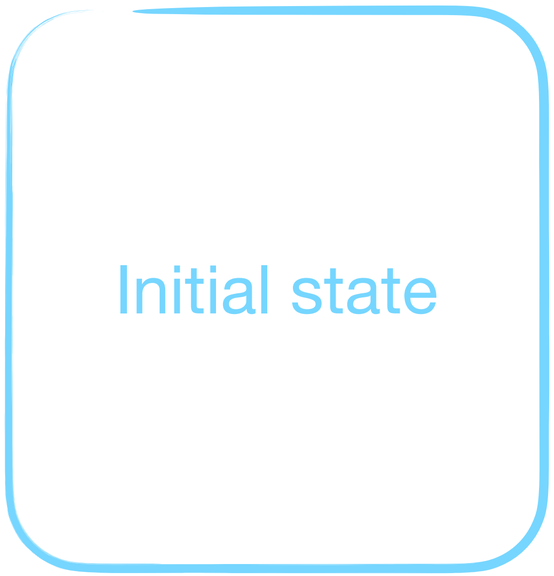
Markov chain ingredients



Where you can go

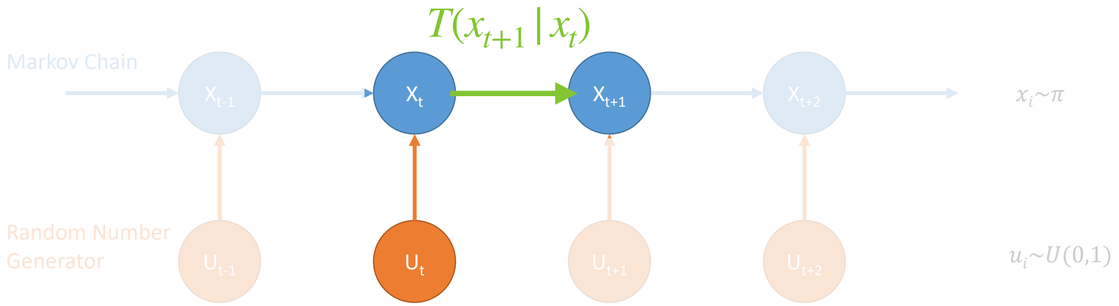
x_t

What you can do



Starting point

Transition matrix



$T(x'|x)$: how to evolve the state

- Discrete x: transition matrix
- Continuous x: transition kernel

$\pi^{(t)}(x)$: distribution over states at time t

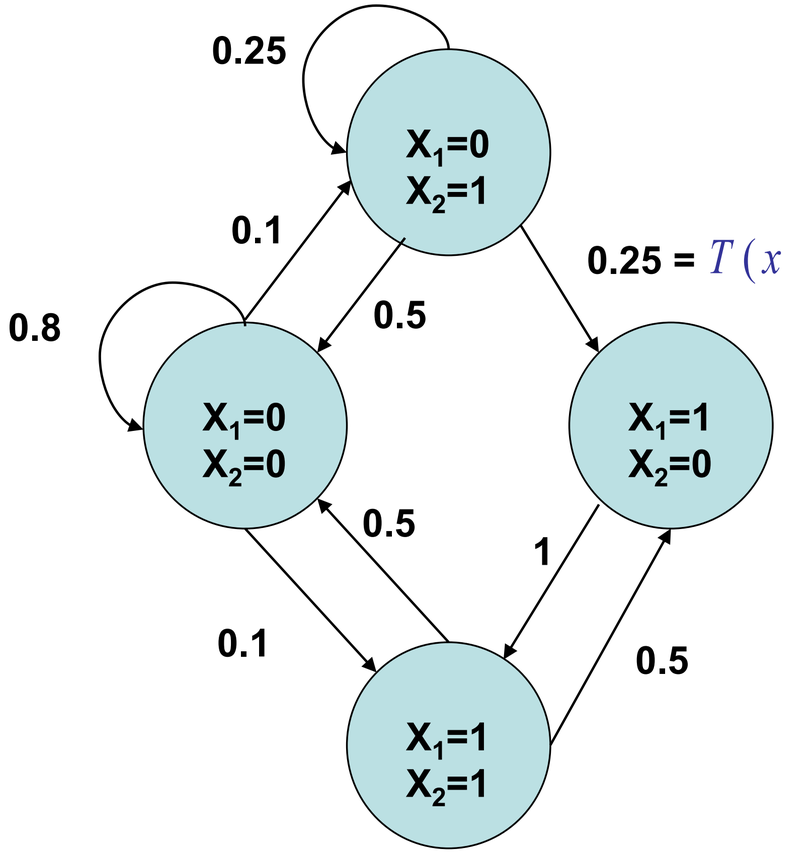
- Need new variable because initially (t=0), not the same as the desired distribution P(x)

Can also evolve distributions: $\pi^{(t+1)}(x') = \sum_x T(x'|x)\pi^{(t)}(x)$

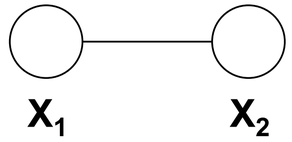
Stationary distribution: does not change with respect to time evolution t.

$$\pi(x') = \sum_x T(x'|x)\pi(x)$$

Markov Chains

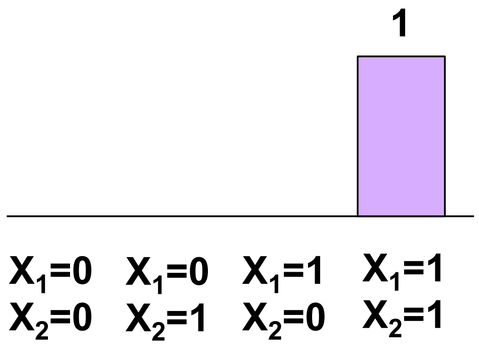


MRF



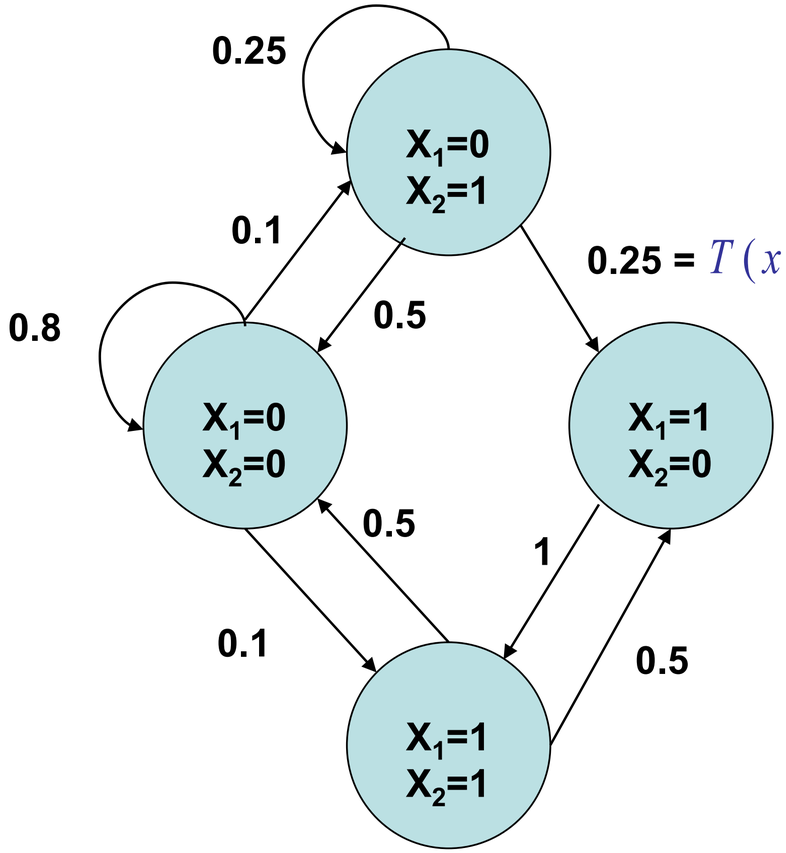
0.25 = $T(x' \mid x)$ = T (x' = (1,0) | x=(0,1))

$\pi^{(0)}(x)$

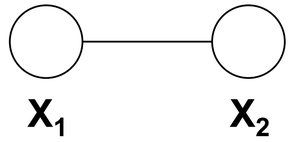


Initialize the simulation in one state $x^{(0)}$

Markov Chains

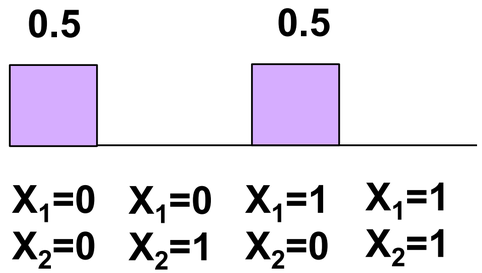


MRF



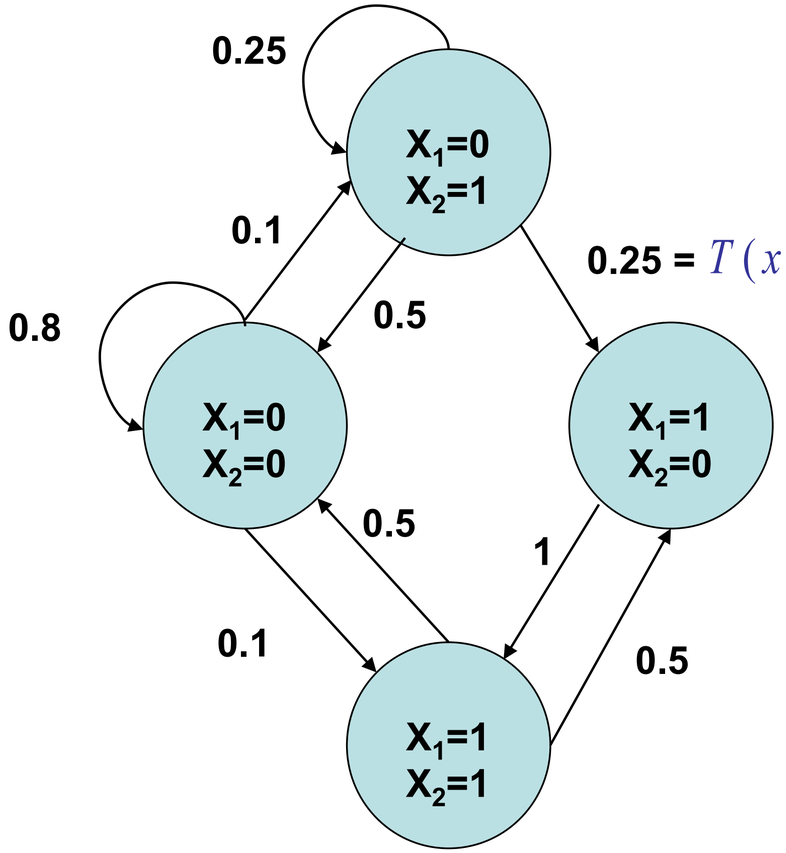
0.25 = $T(x' \mid x)$ = T (x' = (1,0) | x=(0,1))

$\pi^{(1)}(x)$



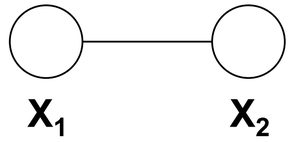
Initialize the simulation in one state $x^{(0)}$

Markov Chains



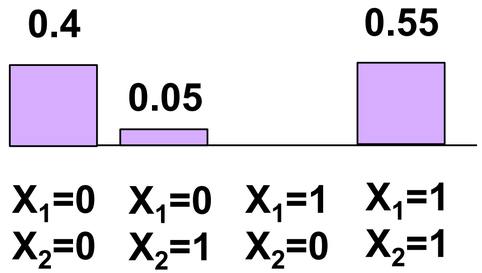
Initialize the simulation in one state $x^{(0)}$

MRF



0.25 = $T(x' \mid x)$ = T (x' = (1,0) | x=(0,1))

$\pi^{(2)}(x)$



stationary if it does not change

Which Markov chains are particularly useful(?)

Irreducible: Can get from any state x to any other state x' in a finite number of timesteps ✈️

- No “unreachable” parts of the state space
- Depends on the transition kernel!

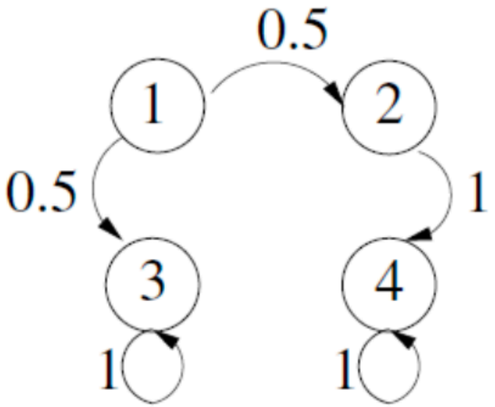
Aperiodic: Can return to any state at any time 🏠

Ergodic: Markov Chain that is irreducible and aperiodic

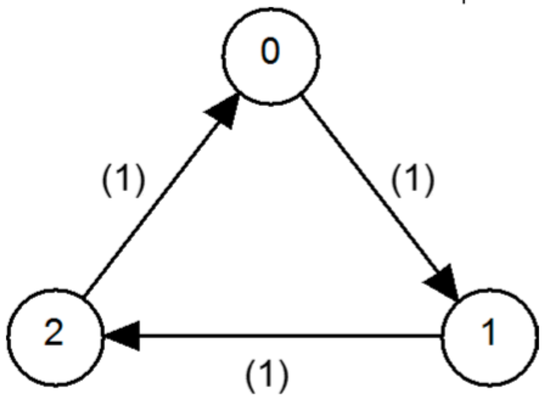
↪ Can reach the stationary distribution $\pi(x)$ no matter the initial dist $\pi^{(0)}(x)$

A.k.a, I can’t initialize in a way that my chain won’t converge!

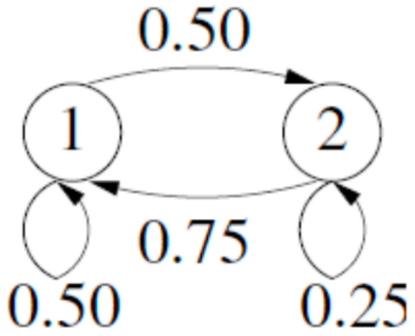
Examples



Reducible.
Limiting distribution depends on initial condition



Irreducible, periodic (each state visited every 3 iterations)
Limiting distribution does not exist



Irreducible, aperiodic.
Unique limiting distribution
P(x) = [0.6, 0.4]

Theorem: If a Markov chain is

1) **ergodic** (irreducible and aperiodic) and
2) satisfies **detailed balance** (is reversible) with respect to p(x),

then p(x) is its unique stationary distribution.

The chain converges to the stationary distribution regardless of where it begins.

Goal: Show Metropolis Hastings works.

- I.e, generates samples from the desired distribution P(x)

How?

1) Show it's reversible
   - Identify the transition matrix
2) Show it is ergodic

   ↪ Need to specify proposal, see tutorial.

Step 2) Reversibility

Reversible (detailed balance): an MC is reversible if there exists a distribution $\pi(x)$ such that the detailed balance condition is satisfied:

$$\pi(x')T(x \mid x') = \pi(x)T(x' \mid x)$$

- Probability of x'→x is the same as x→x'

- $\pi(x)$ is a stationary distribution of the MC! Proof:

$$\pi(x')T(x \mid x') = \pi(x)T(x' \mid x)$$

$$\sum_x \pi(x')T(x \mid x') = \sum_x \pi(x)T(x' \mid x)$$

$$\pi(x')\sum_x T(x \mid x') = \sum_x \pi(x)T(x' \mid x)$$

$$\pi(x') = \sum_x \pi(x)T(x' \mid x)$$

  - The last line is the definition of a stationary distribution!

Step 2) Show MH converges to this stationary dist

Transition matrix:

$$T(x'|x) = Q(x'|x)\ A(x'|x)$$

Proposal Prob of acceptance

$$A(x'|x) = \min\left(1, \frac{P(x')\ Q(x|x')}{P(x)\ Q(x'|x)}\right)$$

If $A(x'|x) < 1$, then $\frac{P(x)Q(x'|x)}{P(x')Q(x|x')} > 1$, so $A(x|x') = 1$

free to multiply by 1

$$A(x'|x) = \frac{P(x')Q(x|x')}{P(x)Q(x'|x)} \implies P(x)\underbrace{Q(x'|x)A(x'|x)}_{T(x'|x)} = P(x')\underbrace{Q(x|x')\ A(x|x')}_{T(x|x')}$$

$$P(x)T(x'|x) = P(x')T(x|x') \longrightarrow \textbf{Detailed balance condition}$$

If ergodic (irreducible & aperiodic), MH algorithm eventually converges to the true distribution!

Practical concerns

A few “hyperparameters” :

- Acceptance rate
- Autocorrelation

What proposal?

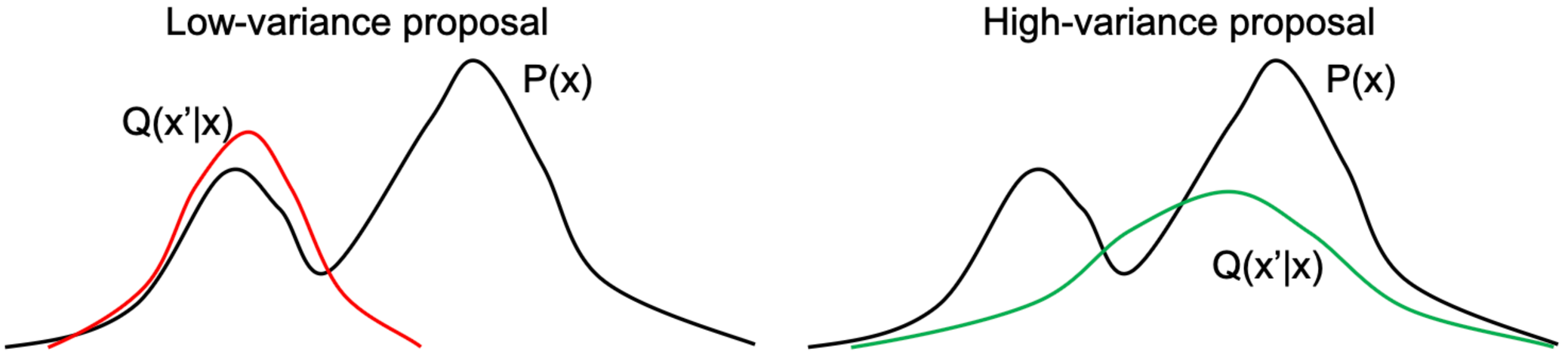
How long to “burn in” 🔥

- Plot sample values vs. time
- Log likelihood vs. time

1. Initialize starting state $x^{(0)}$, set $t=0$
2. Burn-in: while samples have “not converged”
   - $x=x^{(t)}$
   - $t=t+1$
   - sample $x^* \sim Q(x^*|x)$ // draw from proposal
   - sample $u \sim$ Uniform(0,1) // draw acceptance threshold
   - if $u < A(x^*|x) = \min\left(1, \frac{P(x^*)Q(x|x^*)}{P(x)Q(x^*|x)}\right)$
     - $x^{(t)} = x^*$ // transition
   - else
     - $x^{(t)} = x$ // stay in current state
- Take samples from P(x): Reset t=0, for $t=1{:}N$
  - x(t+1) ← Draw sample (x(t))
- (Monte Carlo Estimation using these N final samples)

Function
Draw sample (x(t))

Acceptance rate



Choosing the proposal distribution Q(x' | x) is a trade off

- "Narrow", low-variance proposals have high acceptance, but take many iterations to explore P(x) fully because the proposed x are too close
- Wide, high-variance proposals have the potential to explore much of P(x), but many proposals are rejected which slows down the sampler

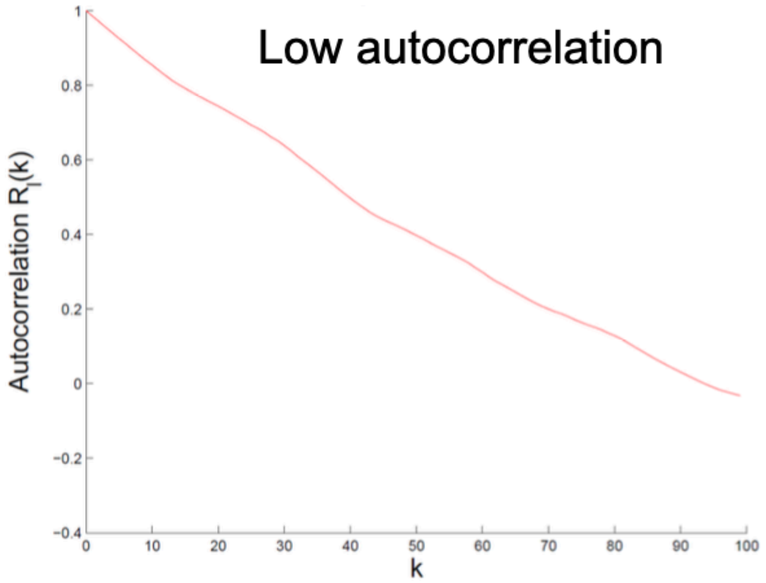
A good Q(x'|x) proposes distant samples x' with a sufficiently high acceptance rate

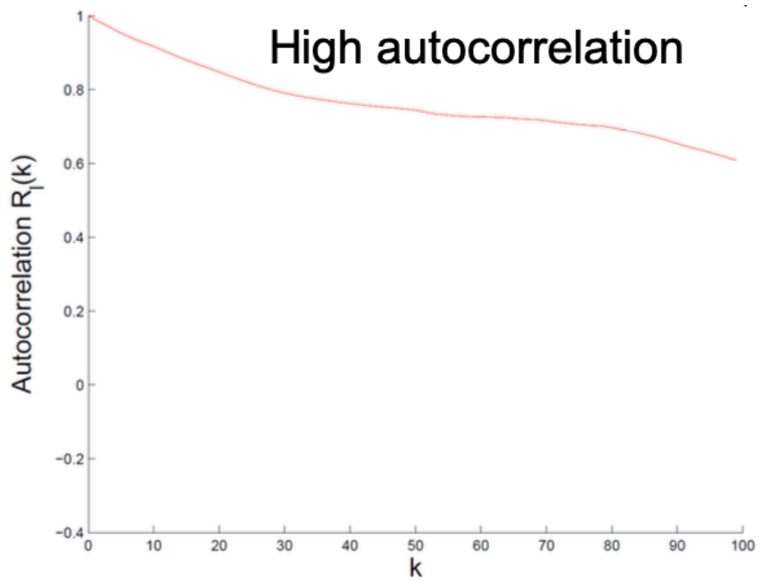
Autocorrelation

MCMC will produces correlated sample

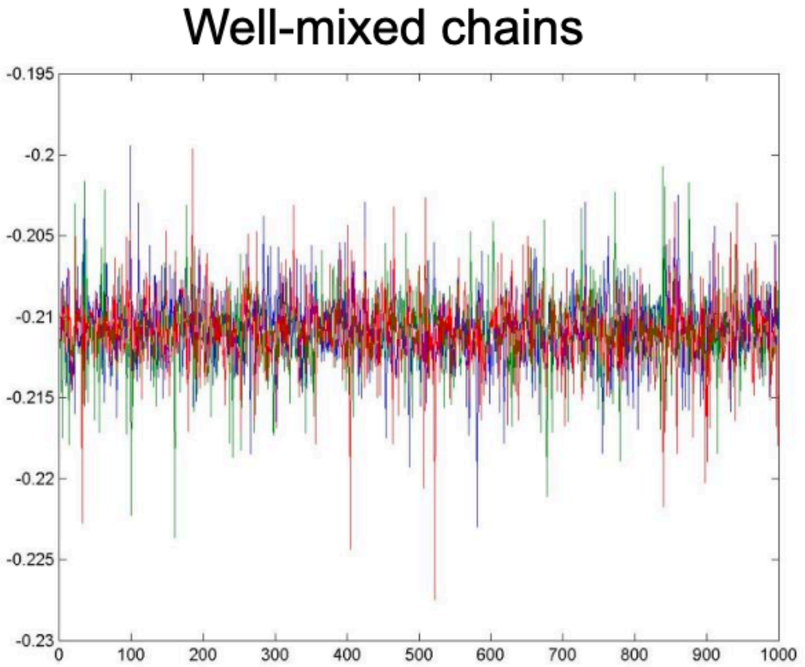
- Reduces the effectiveness sample size

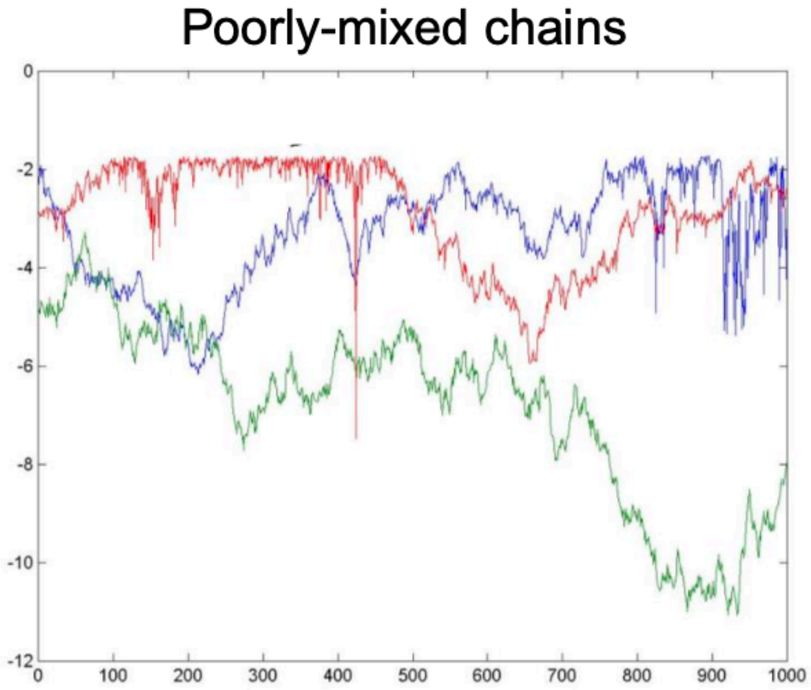
Autocorrelation: $$R_x(k) = \frac{\sum_{t=1}^{n-k}(x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^{n-k}(x_t - \bar{x})^2}$$





Samples vs. time





Monitor convergence by plotting samples from multiple MH runs (chains)

- Well mixed chains (left) -> properly converged
- \Poorly mixed chains (right) -> should continue the burn in

Connection to posteriors

In Bayesian Inference, we want to know properties of the posterior probability:

$$p(\theta|x, M) = \frac{p(x|\theta, M)p(\theta|M)}{p(x|M)}$$

Using Metropolis-(Hastings), we only need the **ratio** for the transition probability from state $\theta_{old} \rightarrow \theta_{new}$:

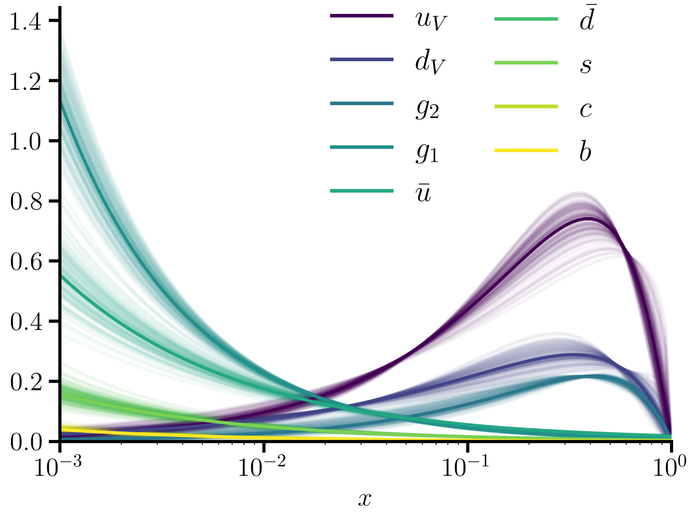
$$\frac{p(\theta_{new}|x, M)}{p(\theta_{old}|x, M)} = \frac{p(x|\theta_{new}, M)p(\theta_{new}|M)}{p(x|\theta_{old}, M)p(\theta_{old}|M)}$$

The evidence cancels out!!! (it is independent of θ)

2401.17729

Ex: Determining the proton's structure

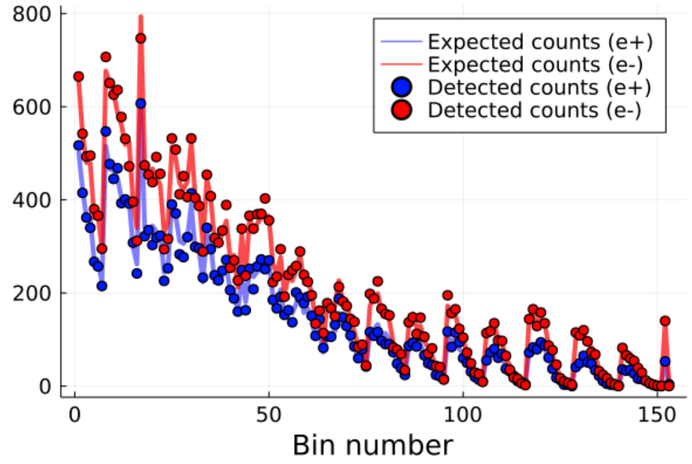
Input: Parton Density Functions, $\theta \in \mathbb{R}^{16}$



Forward Model

Bayesian Inference, what θ?

Input: Parton Density Functions, $\theta \in \mathbb{R}^{16}$



$$p(\theta \mid D) \propto p(D \mid \theta) p(\theta)$$

Sample posterior with Metropolis Hastings

Other variations 

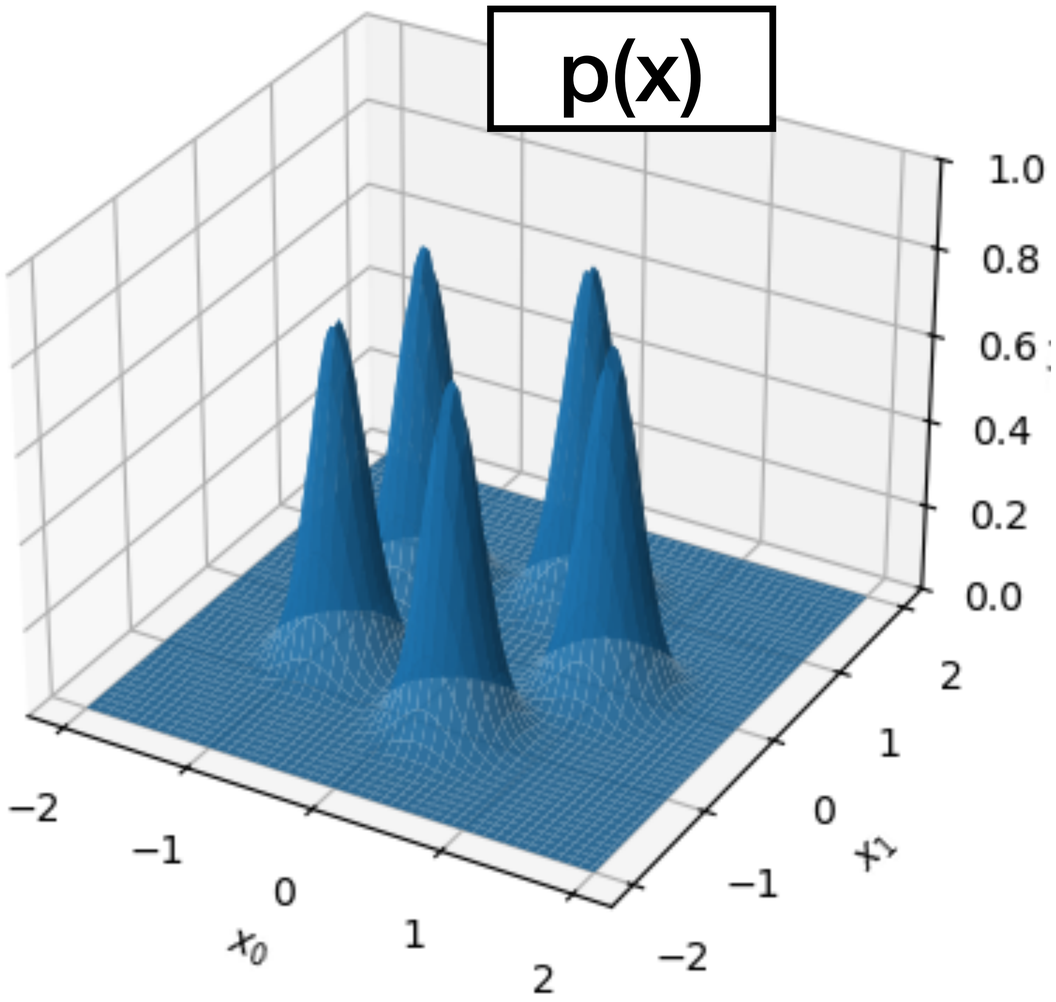
Gibbs Sampling

- Special case of Metropolis Hastings where the next step is always accepted

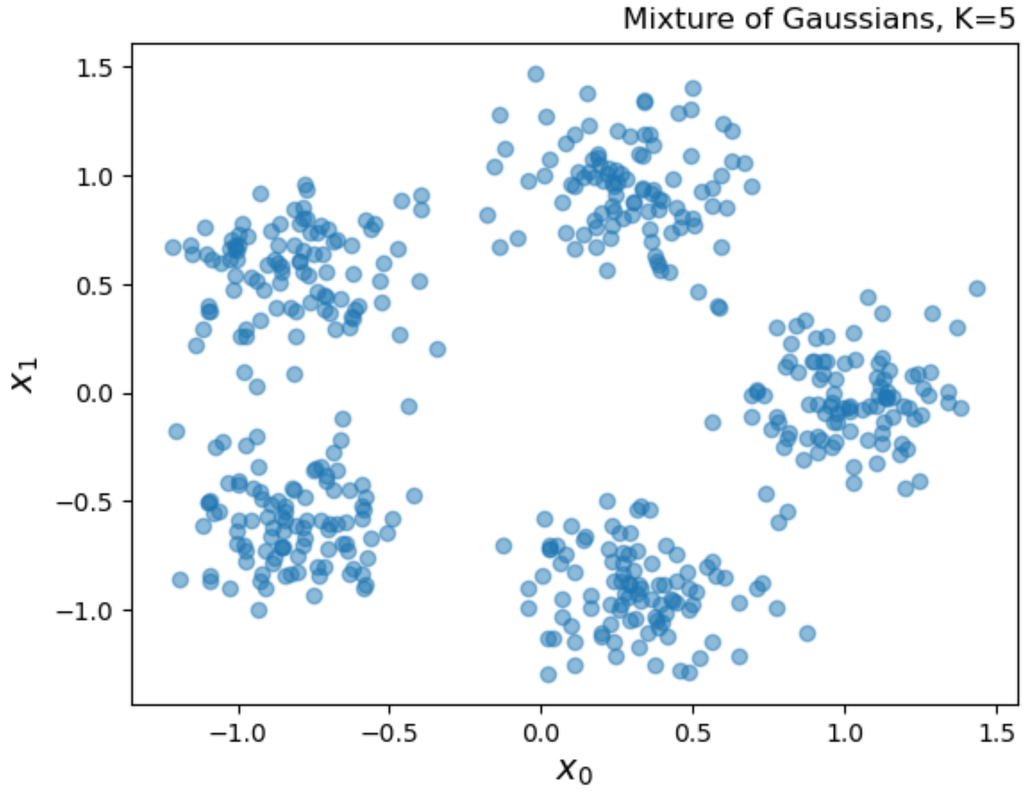
Hamiltonian Monte Carlo

- Uses momentum to navigate more effectively

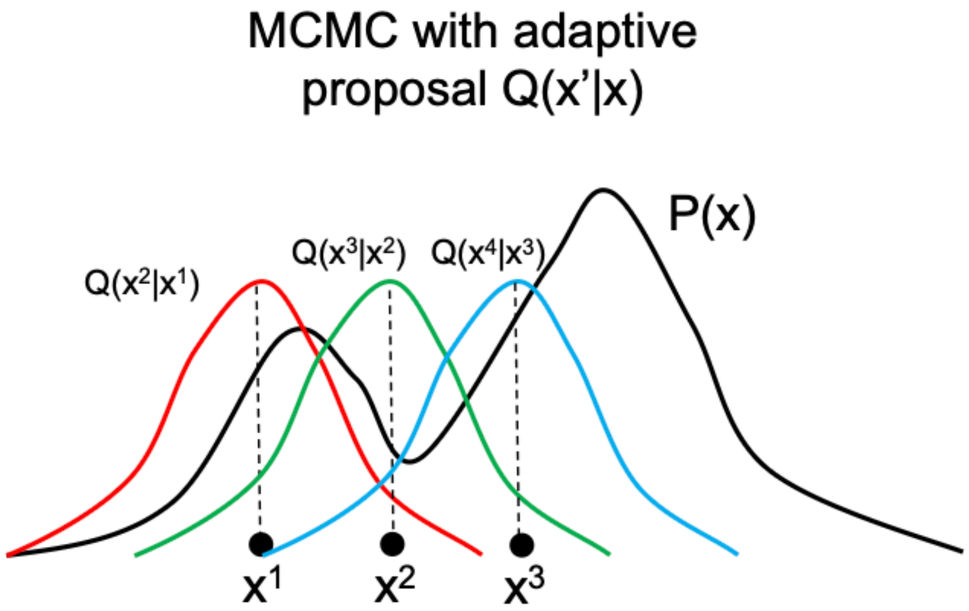
Tutorial: Given a density, can I sample from it?

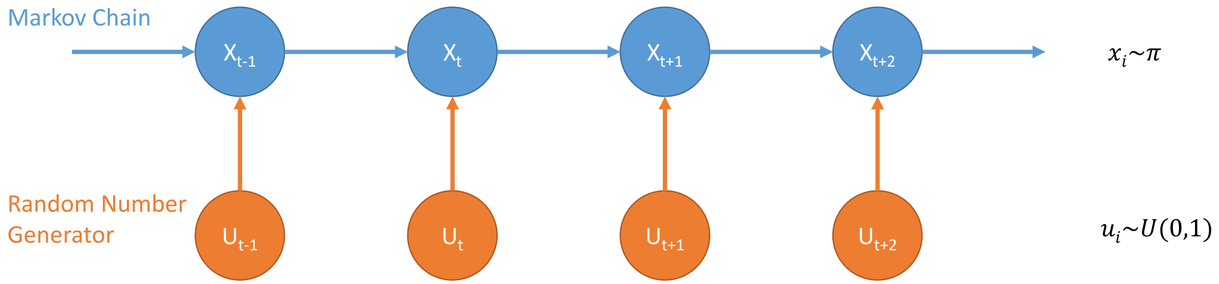


Metropolis-Hastings



In summary





$$P(x_t \mid x_{t-1}, x_{t-2} \ldots, x_1) = P(x_t \mid x_{t-1})$$

Metropolis Hasting converges to the stationary distribtuion π(x) = P(x)

Extra

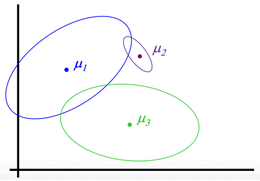
Mixture of Gaussians model

- P(Y): There are k components
- P(X|Y): Each component generates data from a **multivariate Gaussian** with mean μ_i and covariance matrix Σ_i

Each data point is sampled from a ***generative process***:

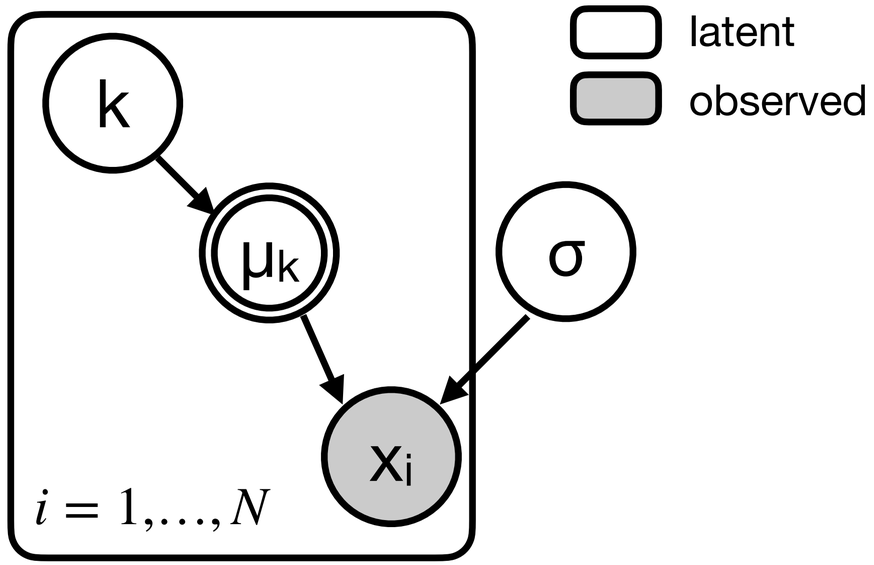
1. Choose component i with probability *P(y=i)*
2. Generate datapoint ~ N(m_i, Σ_i)

Gaussian mixture model (GMM)



$$\{0,1,2,3,4\}$$

$$\mu_k = \begin{pmatrix} \cos\frac{2\pi k}{5} \\ \sin\frac{2\pi k}{5} \end{pmatrix}$$



$$\mathcal{N}(\mu_k, \sigma^2)$$

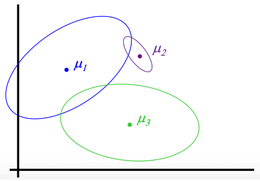
Mixture of Gaussians model

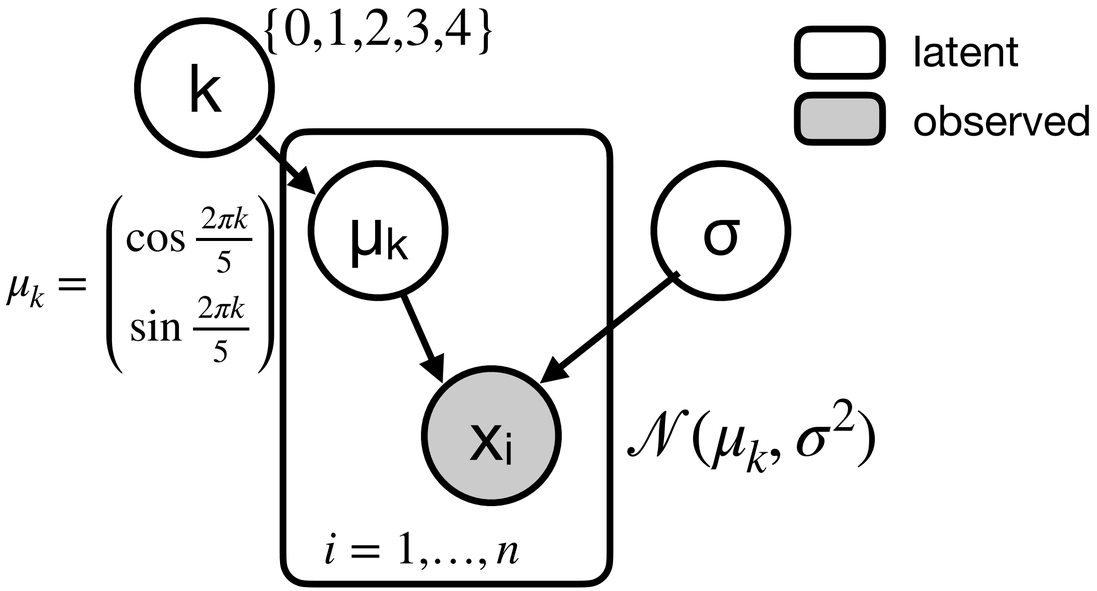
- P(Y): There are k components
- P(X|Y): Each component generates data from a **multivariate Gaussian** with mean μ_i and covariance matrix Σ_i

Each data point is sampled from a ***generative process***:

1. Choose component i with probability *P(y=i)*
2. Generate datapoint ~ N(m_i, Σ_i)

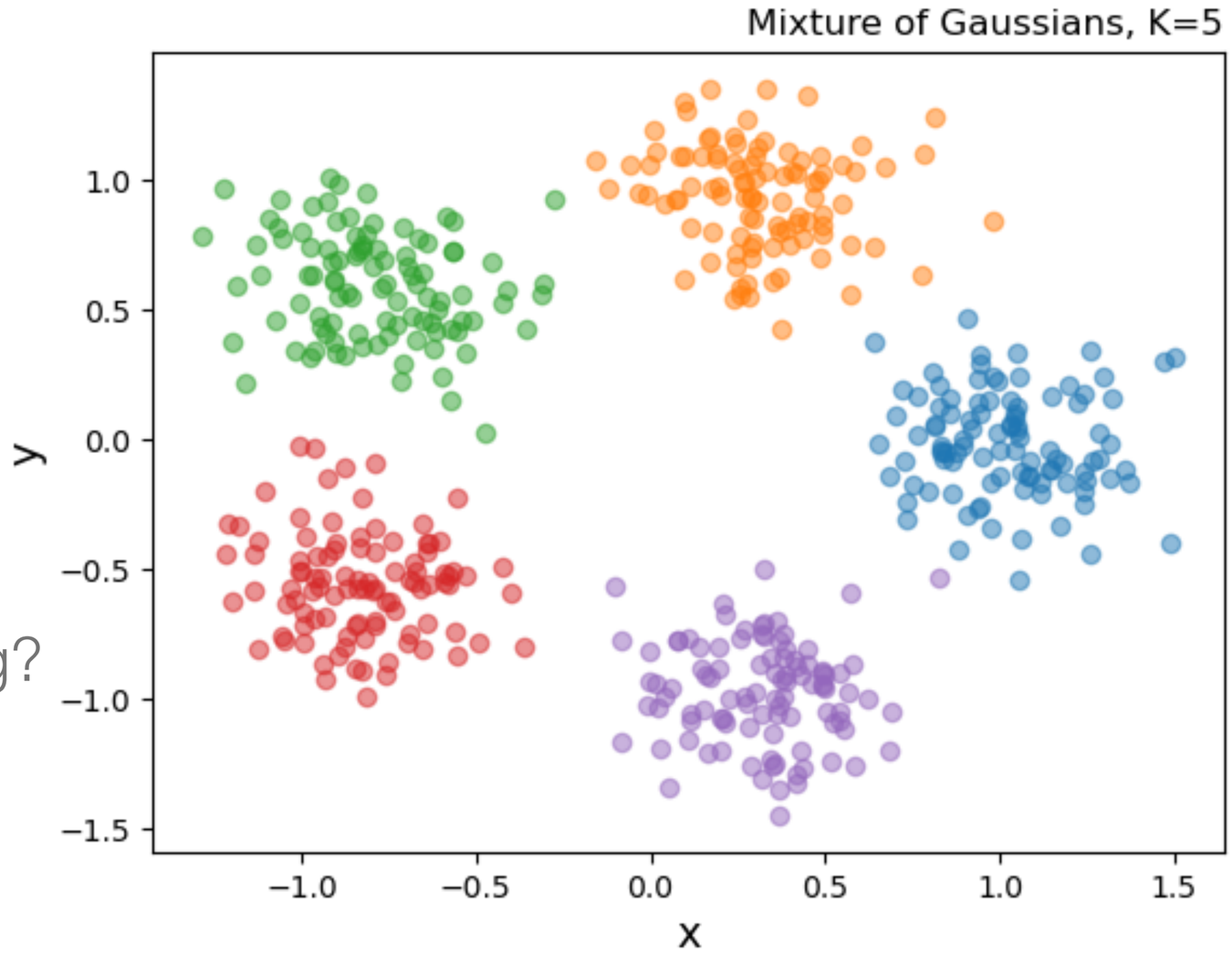
Gaussian mixture model (GMM)





Tutorial: Given a density, can I sample from it?

Maybe a better ex would be
p(theta, x) with a signal and bkg?

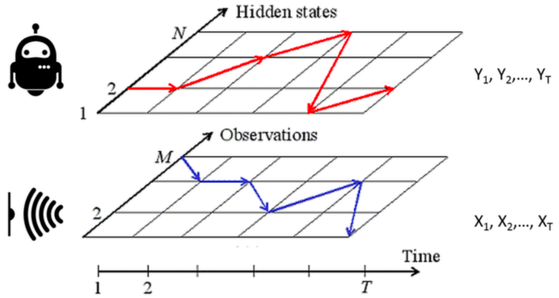


PGM Ex: Hidden markov model

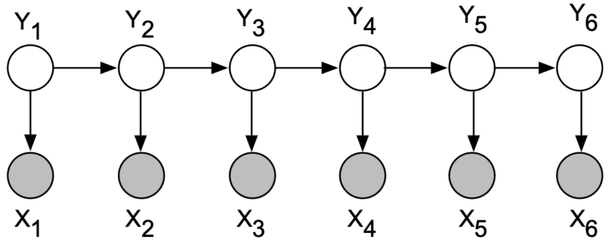
Kalman filter, speech recognition or particle in detector

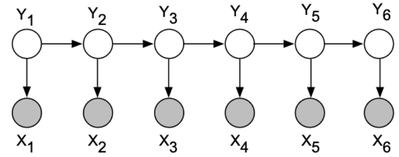
Hidden Markov models

- Frequently used for speech recognition, part-of-speech tagging, Kalman filtering. Hidden variables (Y) and Observations (X)



- Model as a Bayes Net:





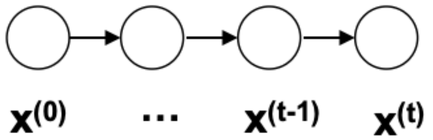
- Joint distribution factors as:

$$p(\mathbf{y}, \mathbf{x}) = p(y_1)p(x_1 \mid y_1) \prod_{t=2}^{T} p(y_t \mid y_{t-1})p(x_t \mid y_t)$$

  - $p(y_1)$ is the distribution for the starting state
    - Example: $p(Y_1 = 2) = 1.0$
  - $p(y_t \mid y_{t-1})$ is the *transition* probability between any two states
    - Example: $p(Y_t = k + 1 | Y_{t-1} = k) = 0.5 = p(Y_t = k - 1 | Y_{t-1} = k)$
  - $p(x_t \mid y_t)$ is the *emission* probability
    - Example: $p(X_t = k | Y_t = k) = 0.99$, $p(X_t = 0 | Y_t = k) = 0.01$
- What are the conditional independencies here? $X_1 \perp X_6$? $Y_1 \perp \{Y_3, \ldots, Y_6\} \mid Y_2$?

Markov Chains

- A Markov Chain is a sequence of random variables $x^{(1)}, x^{(2)}, \ldots, x^{(t)}$ with the Markov Property



$$P(x^{(t)} = x \mid x^{(1)}, \ldots, x^{(t-1)}) = P(x^{(t)} = x \mid x^{(t-1)})$$

  - $P(x^{(t)} = x \mid x^{(t-1)})$ is known as the transition kernel
  - The next state depends only on the preceding state – recall HMMs!
  - Note: the r.v.s $x^{(i)}$ can be vectors
    - We define $x^{(t)}$ to be the t-th sample of all variables in a graphical model
    - $x^{(t)}$ represents the entire state of the graphical model at time t

- We study homogeneous Markov Chains, in which the transition kernel $P(x^{(t)} = x' \mid x^{(t-1)} = x)$ is fixed with time
  - To emphasize this, we will call the kernel $T(x' \mid x)$, where x is the previous state and x' is the next state