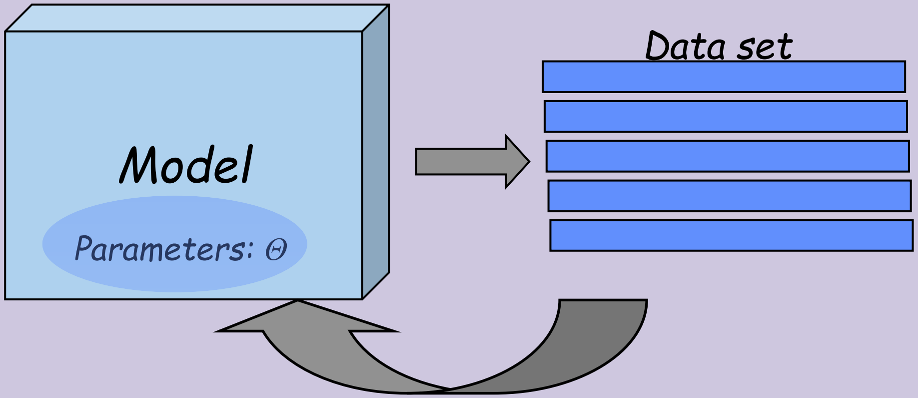
Markov Chain Monte Carlo

Markov Chains

Statistical Parameter Estimation

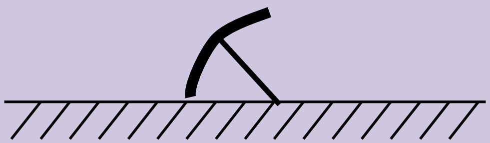
Reminder

- The basic paradigm:

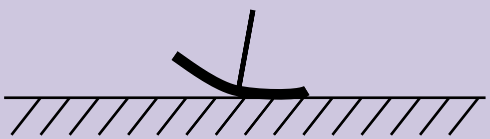


- MLE / bayesian approach

- Input data: series of observations $X_1, X_2 \ldots X_t$
  - We assumed observations were i.i.d (independent identical distributed)



Heads - $P(H)$

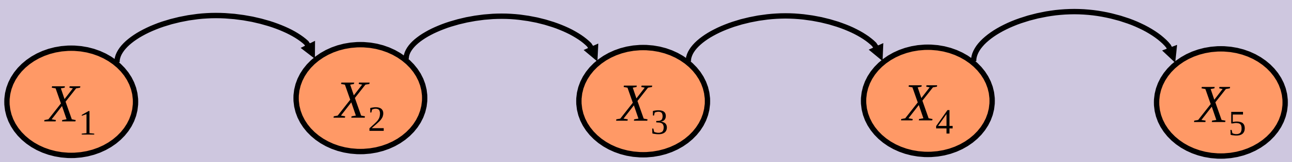


Tails - $1\text{-}P(H)$

Markov Process

- Markov Property: The state of the system at time $t+1$ depends only on the state of the system at time t

$$\boxed{\Pr\left[X_{t+1} = x_{t+1} \mid X_1 \cdots X_t = x_1 \cdots x_t\right] = \Pr\left[X_{t+1} = x_{t+1} \mid X_t = x_t\right]}$$



- Stationary Assumption: Transition probabilities are independent of time (t)

$$\boxed{\Pr\left[X_{t+1} = b \mid X_t = a\right] = p_{ab}}$$

Bounded memory transition model

Markov Process

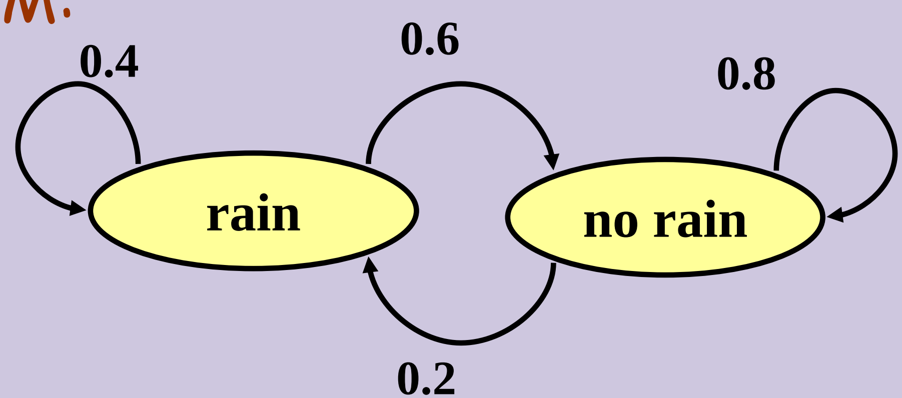
Simple Example

Weather:

- raining today ⇨ 40% rain tomorrow
  ⇨ 60% no rain tomorrow

- not raining today ⇨ 20% rain tomorrow
  ⇨ 80% no rain tomorrow

Stochastic FSM:



Markov Process
Simple Example

Weather:

- **raining today** ⇒ **40% rain tomorrow**
  ⇒ **60% no rain tomorrow**

- **not raining today** ⇒ **20% rain tomorrow**
  ⇒ **80% no rain tomorrow**

The transition matrix:

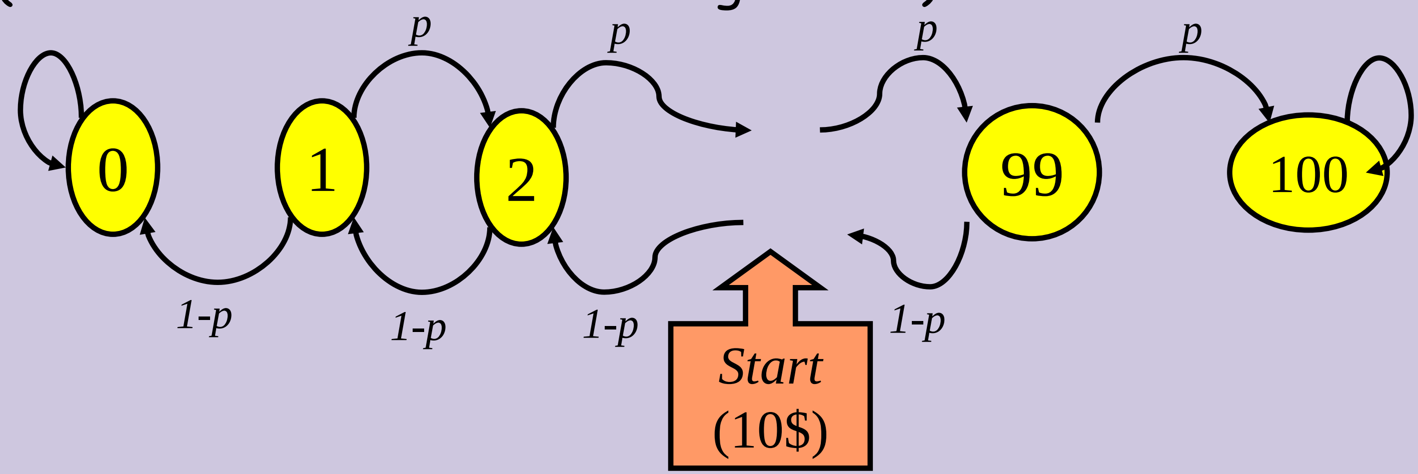
$$P = \begin{pmatrix} 0.4 & 0.6 \\ 0.2 & 0.8 \end{pmatrix}$$

- Stochastic matrix:
  Rows sum up to 1
- Double stochastic matrix:
  Rows and columns sum up to 1

Markov Process
Gambler's Example

- Gambler starts with $10
- At each play we have one of the following:
  - Gambler wins $1 with probability p
  - Gambler looses $1 with probability 1-p
- Game ends when gambler goes broke, or gains a fortune of $100

  (Both 0 and 100 are absorbing states)

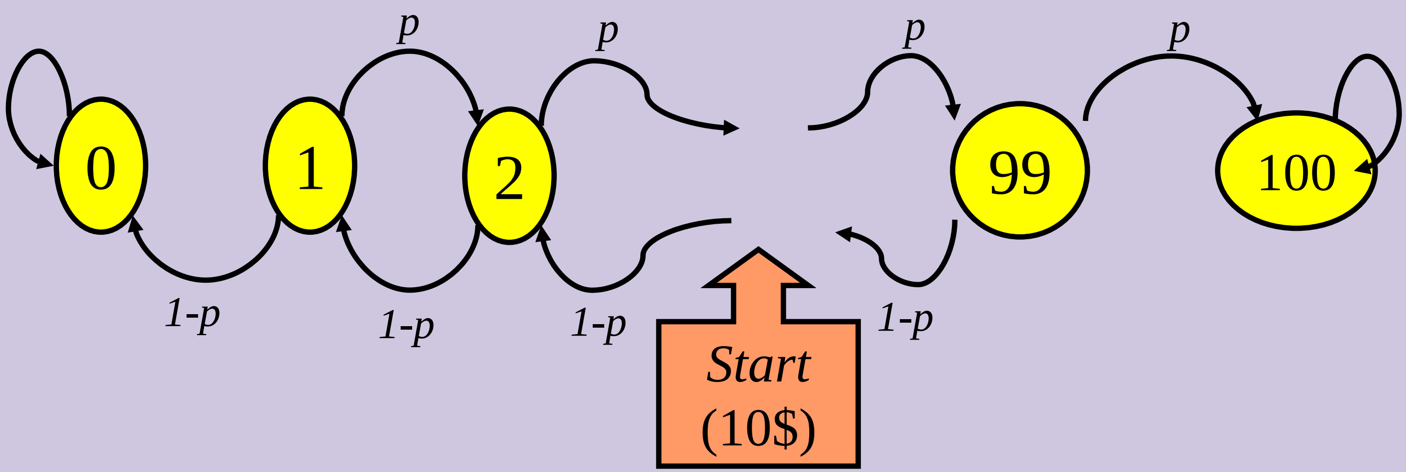


Markov Process

- **Markov process** - described by a stochastic FSM
- **Markov chain** - a random walk on this graph

(distribution over paths)

- Edge-weights give us $\Pr\left[X_{t+1}=b \mid X_t=a\right]=p_{ab}$
- We can ask more complex questions, like $\Pr\left[X_{t+2} = a \mid X_t = b\right]=?$



Markov Process

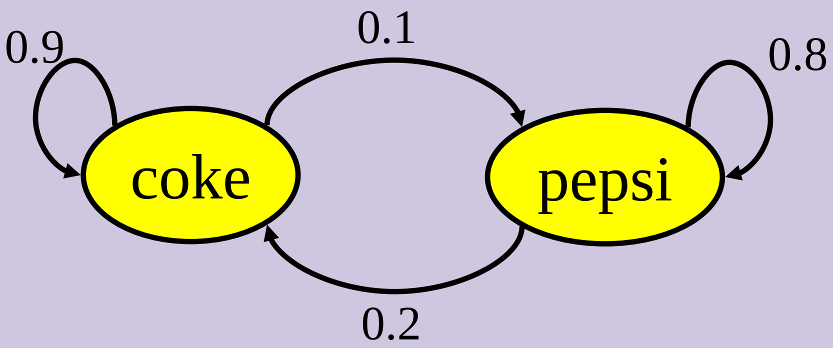
Coke vs. Pepsi Example

• Given that a person's last cola purchase was Coke, there is a 90% chance that his next cola purchase will also be Coke.

• If a person's last cola purchase was Pepsi, there is an 80% chance that his next cola purchase will also be Pepsi.

transition matrix:

$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix}$$



Markov Process

Coke vs. Pepsi Example (cont)

Given that a person is currently a Pepsi purchaser, what is the probability that he will purchase Coke two purchases from now?

Pr[Pepsi→?→Coke] =

Pr[Pepsi→Coke→Coke] + Pr[Pepsi→ Pepsi →Coke] =

0.2 * 0.9 + 0.8 * 0.2 = 0.34

$$P^2 = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix} = \begin{bmatrix} 0.83 & 0.17 \\ 0.34 & 0.66 \end{bmatrix}$$

Pepsi → ? ? → Coke

Markov Process

Coke vs. Pepsi Example (cont)

Given that a person is currently a Coke purchaser, what is the probability that he will purchase Pepsi **three** purchases from now?

$$P^3 = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix} \begin{bmatrix} 0.83 & 0.17 \\ 0.34 & 0.66 \end{bmatrix} = \begin{bmatrix} 0.781 & 0.219 \\ 0.438 & 0.562 \end{bmatrix}$$

Markov Process

Coke vs. Pepsi Example (cont)

- Assume each person makes one cola purchase per week
- Suppose 60% of all people now drink Coke, and 40% drink Pepsi
- What fraction of people will be drinking Coke three weeks from now?

$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix} \qquad P^3 = \begin{bmatrix} 0.781 & 0.219 \\ 0.438 & 0.562 \end{bmatrix}$$

$$\Pr[X_3 = \text{Coke}] = 0.6 * 0.781 + 0.4 * 0.438 = 0.6438$$

Q_i - the distribution in week i

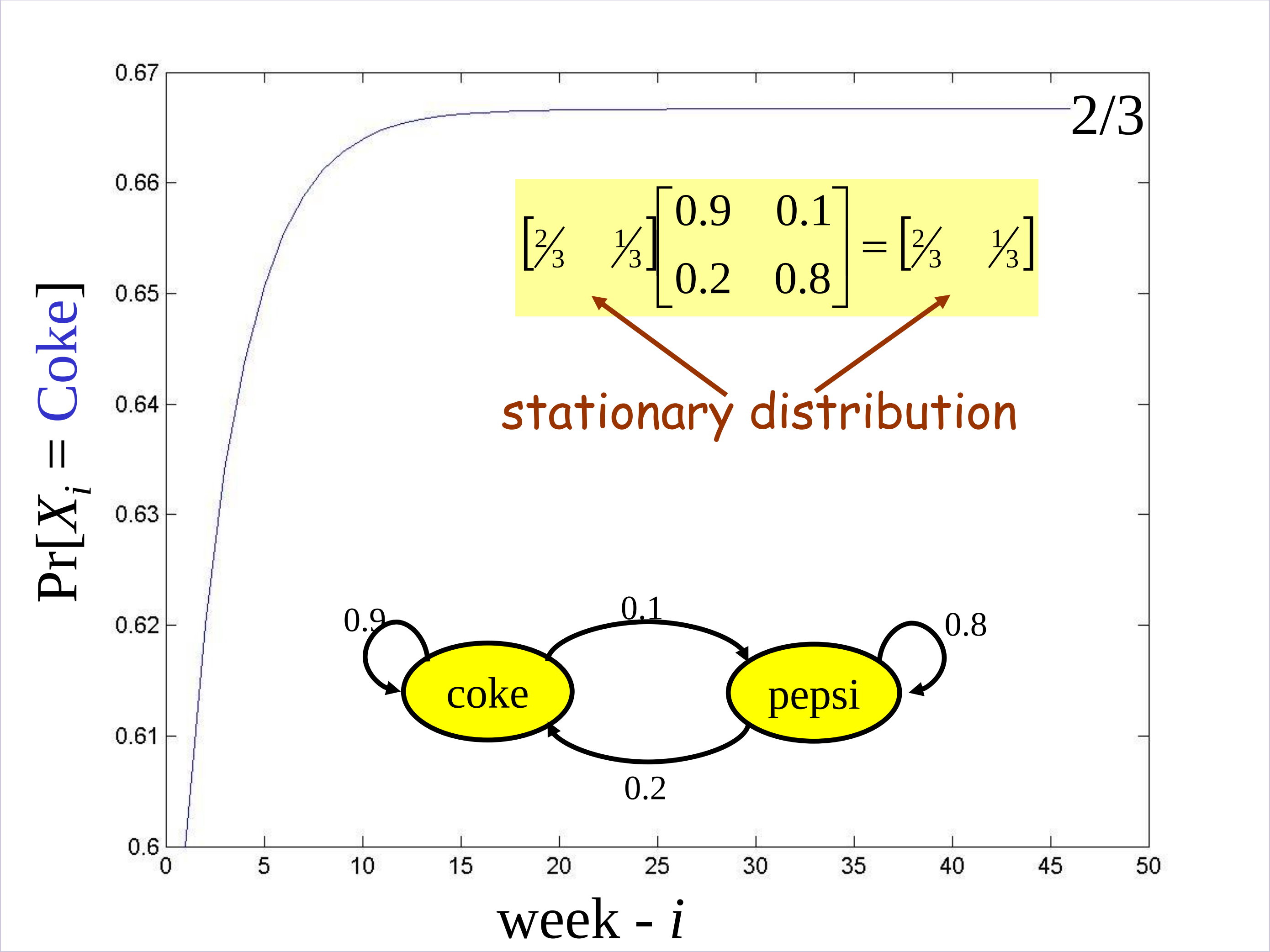
$Q_0 = (0.6, 0.4)$ - initial distribution

$Q_3 = Q_0 * P^3 = (0.6438, 0.3562)$

Markov Process

Coke vs. Pepsi Example (cont)

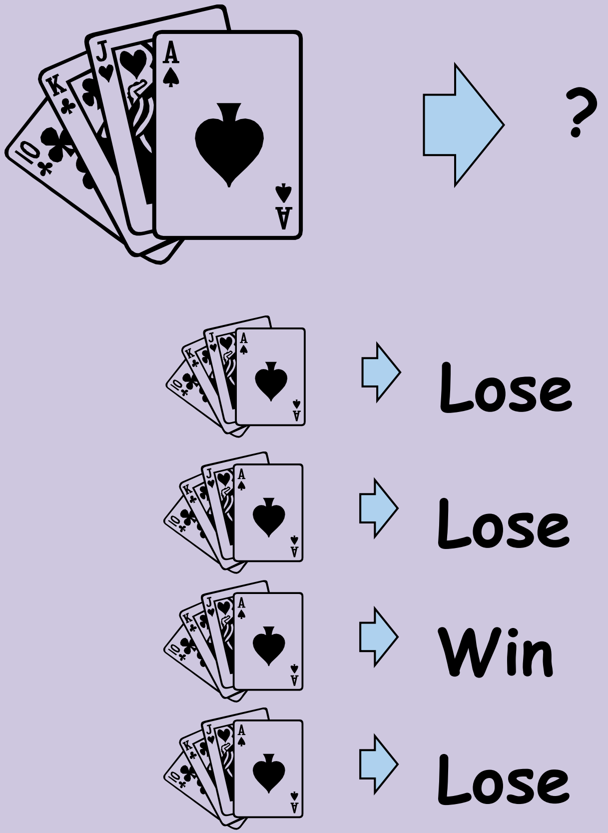
Simulation:



Markov Chain Monte Carlo

Monte Carlo principle

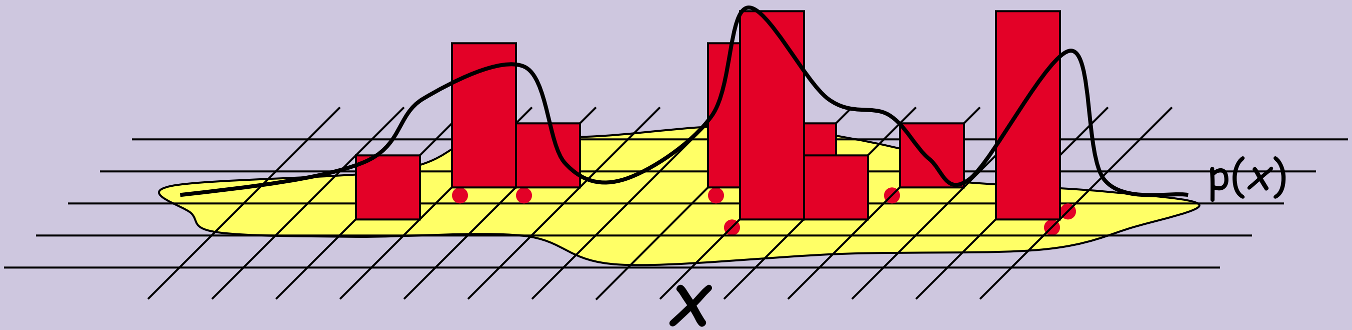
- Consider the game of solitaire: what’s the chance of winning with a properly shuffled deck?
- Hard to compute analytically because winning or losing depends on a complex procedure of reorganizing cards
- Insight: why not just *play a few hands*, and see empirically how many do in fact win?
- More generally, can approximate a probability density function using only samples from that density



Chance of winning is 1 in 4!

Monte Carlo principle

- Given a very large set *X* and a distribution p(*x*) over it
- We draw i.i.d. a set of N samples
- We can then approximate the distribution using these samples



$$\mathrm{p}_N(x) = \frac{1}{N}\sum_{i=1}^{N} 1(x^{(i)} = x) \underset{N\to\infty}{\longrightarrow} \mathrm{p}(x)$$

Monte Carlo principle

- We can also use these samples to compute expectations

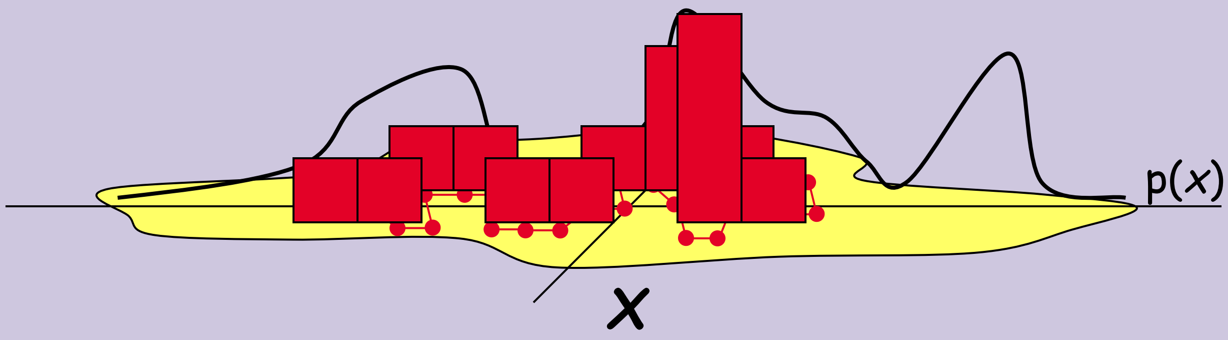
$$E_N(f) = \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \underset{N\to\infty}{\longrightarrow} E(f) = \sum_{x} f(x)\,\mathrm{p}(x)$$

- And even use them to find a maximum

$$\hat{x} = \underset{x^{(i)}}{\arg\max}[\mathrm{p}(x^{(i)})]$$

Markov chain Monte Carlo

- Recall again the set X and the distribution p(x) we wish to sample from
- Suppose that it is hard to sample p(x) but that it is possible to "walk around" in X using only local state transitions
- Insight: we can use a "random walk" to help us draw random samples from p(x)



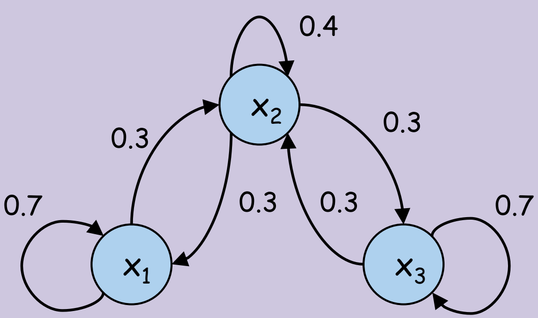
Markov chains

- Markov chain on a space ***X*** with transitions ***T*** is a random process (infinite sequence of random variables)
  $(x^{(0)}, x^{(1)},\ldots x^{(t)},\ldots)$ in $\boldsymbol{X}^{\infty}$ that satisfy

$$\mathrm{p}(x^{(t)} \mid x^{(t-1)},\ldots,x^{(1)}) = \mathrm{T}(x^{(t-1)}, x^{(t)})$$

- That is, the probability of being in a particular state at time *t* given the state history depends only on the state at time *t*-1
- If the transition probabilities are fixed for all *t*, the chain is considered *homogeneous*

$$T = \begin{pmatrix} 0.7 & 0.3 & 0 \\ 0.3 & 0.4 & 0.3 \\ 0 & 0.3 & 0.7 \end{pmatrix}$$



Markov Chains for sampling

- In order for a Markov chain to useful for sampling p(*x*), we require that for any starting state $x^{(1)}$

$$\mathrm{p}_{x^{(1)}}^{(t)}(x) \underset{t\to\infty}{\longrightarrow} \mathrm{p}(x)$$

- Equivalently, the **stationary distribution** of the Markov chain must be p(*x*)

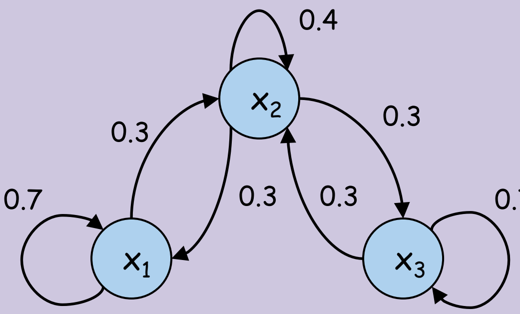
$$[\mathrm{p}\mathbf{T}](x) = \mathrm{p}(x)$$

- If this is the case, we can start in an arbitrary state, use the Markov chain to do a random walk for a while, and stop and output the current state $x^{(t)}$
- The resulting state will be sampled from p(*x*)!

Stationary distribution

- Consider the Markov chain given above:

$$T=\begin{pmatrix} 0.7 & 0.3 & 0 \\ 0.3 & 0.4 & 0.3 \\ 0 & 0.3 & 0.7 \end{pmatrix}$$



- The stationary distribution is

$$\begin{bmatrix} 0.33 & 0.33 & 0.33 \end{bmatrix} \times \begin{pmatrix} 0.7 & 0.3 & 0 \\ 0.3 & 0.4 & 0.3 \\ 0 & 0.3 & 0.7 \end{pmatrix} = \begin{bmatrix} 0.33 & 0.33 & 0.33 \end{bmatrix}$$

- Some samples:

1,1,2,3,2,1,2,3,3,**2**
1,2,2,1,1,2,3,3,3,**3**
1,1,1,2,3,2,2,1,1,**1**
1,2,3,3,3,2,1,2,2,**3**
1,1,2,2,2,3,3,2,1,**1**
1,2,2,2,3,3,3,2,2,**2**

Empirical Distribution:

$$\begin{bmatrix} 0.33 & 0.33 & 0.33 \end{bmatrix}$$

Ergodicity

- Claim: To ensure that the chain converges to a unique stationary distribution the following conditions are sufficient:
  - *Irreducibility*: every state is eventually reachable from any start state; for all *x*, *y* in ***X*** there exists a *t* such that $$\mathrm{p}_x^{(t)}(y) > 0$$
  - *Aperiodicity*: the chain doesn't get caught in cycles; for all *x*, *y* in ***X*** it is the case that $$\gcd\{t : \mathrm{p}_x^{(t)}(y) > 0\} = 1$$
- The process is *ergodic* if it is both irreducible and aperiodic
- This claim is easy to prove, but involves eigenstuff!

Markov Chains for sampling

- Claim: To ensure that the stationary distribution of the Markov chain is p(x) it is sufficient for p and *T* to satisfy the *detailed balance* (*reversibility*) condition:

$$\mathrm{p}(x)T(x,y) = \mathrm{p}(y)T(y,x)$$

- Proof: for all *y* we have

$$[\mathrm{p}\mathbf{T}](y) = \sum_x \mathrm{p}(x)T(x,y) = \sum_x \mathrm{p}(y)T(y,x) = \mathrm{p}(y)\sum_x T(y,x) = \mathrm{p}(y)$$

- And thus p must be a stationary distribution of *T*

Metropolis algorithm

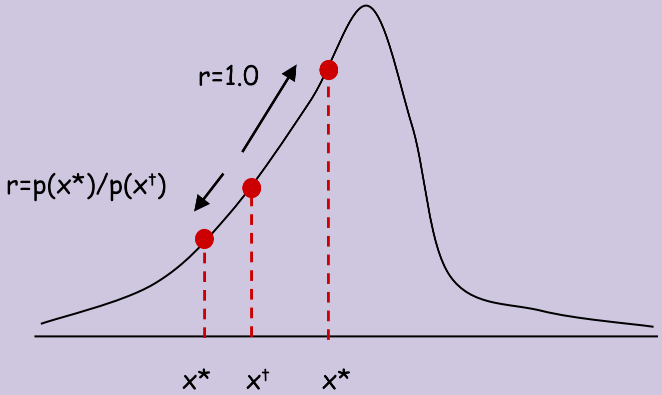
- How to pick a suitable Markov chain for our distribution?
- Suppose our distribution p(x) is easy to sample, and easy to compute *up to a normalization constant*, but hard to compute exactly
  - e.g. a Bayesian posterior $P(M|D) \propto P(D|M)P(M)$
- We define a Markov chain with the following process:
  - Sample a candidate point x* from a *proposal distribution* $q(x^*|x^{(t)})$ which is *symmetric*: $q(x|y)=q(y|x)$
  - Compute the *importance ratio* (this is easy since the normalization constants cancel)

$$r = \frac{\mathrm{p}(x^*)}{\mathrm{p}(x^{(t)})}$$

  - With probability min(*r*,1) transition to x*, otherwise stay in the same state

Metropolis intuition

- Why does the Metropolis algorithm work?
  - Proposal distribution can propose anything it likes (as long as it can jump back with the same probability)
  - Proposal is always accepted if it’s jumping to a more likely state
  - Proposal accepted with the importance ratio if it’s jumping to a less likely state
- The acceptance policy, combined with the reversibility of the proposal distribution, makes sure that the algorithm explores states in proportion to p(*x*)!



Metropolis convergence

- Claim: The Metropolis algorithm converges to the target distribution p(*x*).
- Proof: It satisfies detailed balance

  For all *x*,*y* in ***X***, wlog assuming p(*x*)<p(*y*), then

$$T(x,y) = q(y \mid x)$$

candidate is always accepted, since the r= 1

$$T(y,x) = q(x \mid y)\frac{\mathrm{p}(x)}{\mathrm{p}(y)}$$

Since, w generate x with prob q(x|y) and accept with prob r = the ratio < 1.

q is symmetric

Hence:

$$\mathrm{p}(x)T(x,y) = \mathrm{p}(x)q(y \mid x) = \mathrm{p}(x)q(x \mid y)$$

$$= \mathrm{p}(y)q(x \mid y)\frac{\mathrm{p}(x)}{\mathrm{p}(y)} = \mathrm{p}(y)T(y,x)$$

Metropolis-Hastings

- The symmetry requirement of the Metropolis proposal distribution can be hard to satisfy
- Metropolis-Hastings is the natural generalization of the Metropolis algorithm, and the most popular MCMC algorithm
- We define a Markov chain with the following process:
  - Sample a candidate point x^* from a proposal distribution $q(x^*|x^{(t)})$ which is not necessarily symmetric
  - Compute the importance ratio:

$$r = \frac{p(x^*)\,q(x^{(t)} \mid x^*)}{p(x^{(t)})\,q(x^* \mid x^{(t)})}$$

  - With probability min(r,1) transition to x*, otherwise stay in the same state $x^{(t)}$

MH convergence

- Claim: The Metropolis-Hastings algorithm converges to the target distribution p(x).
- Proof: It satisfies detailed balance

For all x,y in $\boldsymbol{X}$, wlog assume p(x)q(y|x)<p(y)q(x|y), then

$$T(x, y) = q(y \mid x)$$

candidate is always accepted, since r = 1

$$T(y, x) = q(x \mid y)\frac{\mathrm{p}(x)q(y \mid x)}{\mathrm{p}(y)q(x \mid y)}$$

Since, w generate x with prob q(x|y) and accept with prob r = the ratio < 1.

Hence:

$$\mathrm{p}(x)T(x, y) = \mathrm{p}(x)q(y \mid x) = \mathrm{p}(x)q(y \mid x)\frac{\mathrm{p}(y)q(x \mid y)}{\mathrm{p}(y)q(x \mid y)}$$

$$= \mathrm{p}(y)q(x \mid y)\frac{\mathrm{p}(x)q(y \mid x)}{\mathrm{p}(y)q(x \mid y)} = \mathrm{p}(y)T(y, x)$$

Gibbs sampling

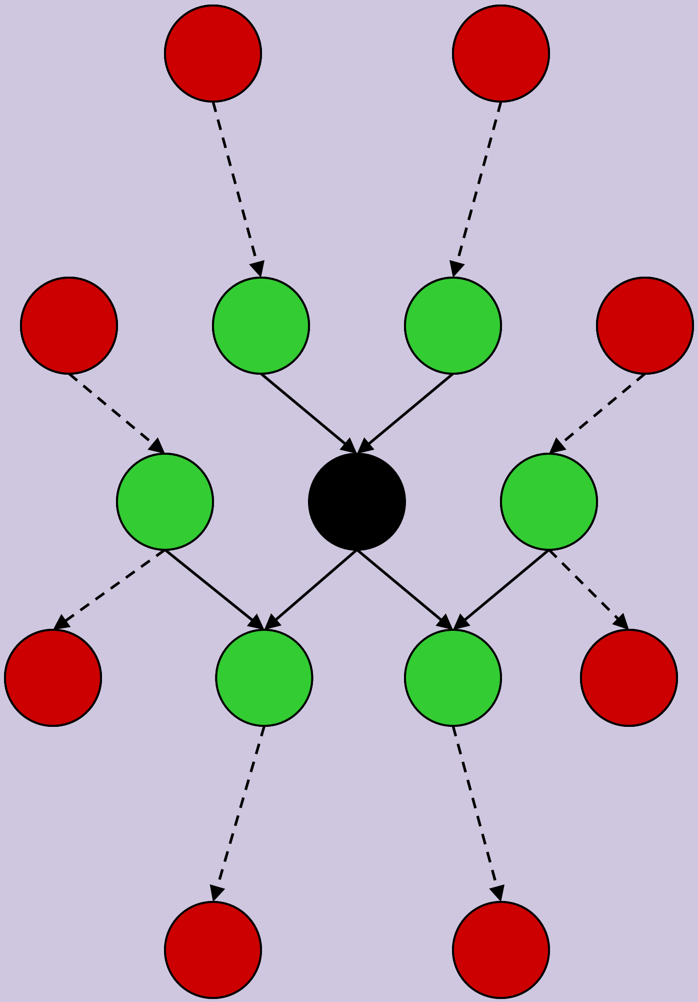
- A special case of Metropolis-Hastings which is applicable to state spaces in which we have a factored state space, and access to the full conditionals:

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

- Perfect for Bayesian networks!
- Idea: To transition from one state (variable assignment) to another,
  - Pick a variable,
  - Sample its value from the conditional distribution
  - That's it!
- We'll show in a minute why this is an instance of MH and thus must be sampling from the full joint

Markov blanket

- Recall that Bayesian networks encode a factored representation of the joint distribution
- Variables are independent of their non-descendents given their parents
- Variables are independent of *everything else in the network* given their *Markov blanket*!
- So, to sample each node, we only need to condition its Markov blanket



$$p(x_j \mid \mathrm{MB}(x_j))$$

Gibbs sampling

- More formally, the proposal distribution is

$$q(x^* \mid x^{(t)}) = \begin{cases} p(x_j^* \mid x_{-j}^{(t)}) & \text{if } x^*_{-j} = x^{(t)}_{-j} \\ 0 & \text{otherwise} \end{cases}$$

- The importance ratio is

$$r = \frac{\mathrm{p}(x^*)\,\mathrm{q}(x^{(t)} \mid x^*)}{\mathrm{p}(x^{(t)})\,\mathrm{q}(x^* \mid x^{(t)})}$$

$$= \frac{\mathrm{p}(x^*)\,\mathrm{p}(x_j^{(t)} \mid x_{-j}^{(t)})}{\mathrm{p}(x^{(t)})\,\mathrm{p}(x_j^* \mid x_{-j}^*)}$$

Dfn of proposal distribution

$$= \frac{\mathrm{p}(x^*)\,\mathrm{p}(x_j^{(t)}, x_{-j}^{(t)})\,\mathrm{p}(x_{-j}^*)}{\mathrm{p}(x^{(t)})\,\mathrm{p}(x_j^*, x_{-j}^*)\,\mathrm{p}(x_{-j}^{(t)})}$$

Dfn of conditional probability

$$= \frac{\mathrm{p}(x_{-j}^*)}{\mathrm{p}(x_{-j}^{(t)})} = 1$$

B/c we didn't change other vars

- So we always accept!

Practical issues

- How many iterations?
- How to know when to stop?
- What’s a good proposal function?