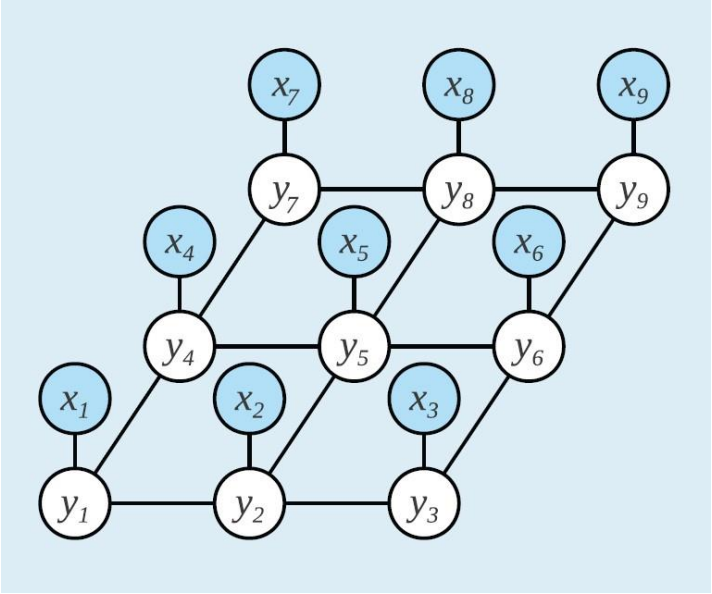
Probabilistic Graphical Models in Bioinformatics

Lecture 15: Particle-based approximate inference



Hesam Montazeri
Department of Bioinformatics, IBB, University of Tehran

Khordad 13, 1398
(June 3, 2019)

Introduction

- Particle-based methods approximate the joint distribution as a set of random samples from the network.
- The term “particle” came from statistical physics.

- Outline
  - Forward sampling
  - Likelihood weighting and importance sampling
  - Markov Chain Monte Carlo

Forward sampling

- We generate random samples $\xi[1], \ldots, \xi[M]$ from the distribution $P(\mathcal{X})$
- We then use the samples to compute the expectation of some target function f.

$$\hat{\boldsymbol{E}}_{\mathcal{D}}(f) = \frac{1}{M} \sum_{m=1}^{M} f(\xi[m]).$$

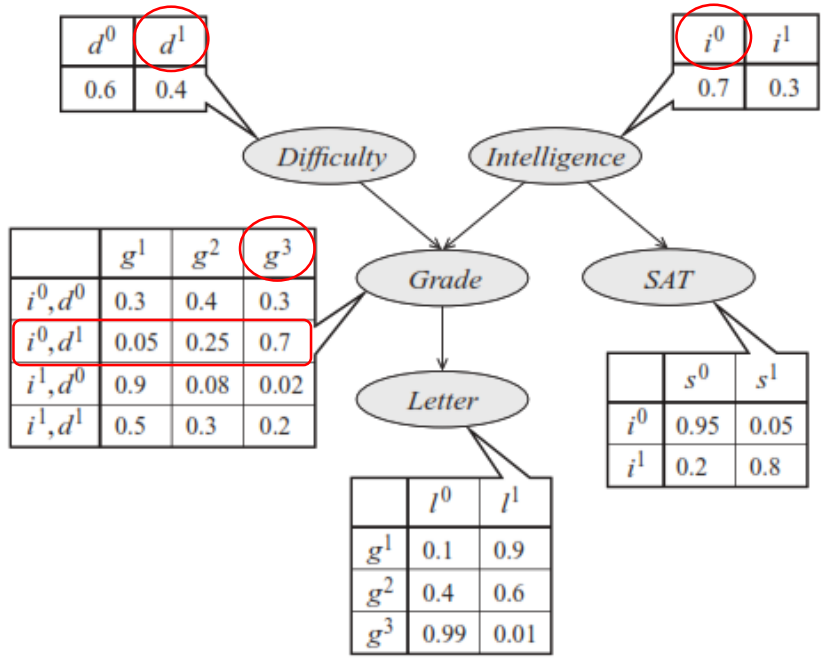
$$\hat{P}_{\mathcal{D}}(\boldsymbol{y}) = \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{I}\{\boldsymbol{y}[m] = \boldsymbol{y}\},$$

$y[m]$ denotes assignment to variables Y in the particles $\xi[m]$

Or simply the fraction of particles with the event y observed.

Forward sampling- sampling from a Bayesian network

- A very simple process
- We sample the nodes in some order consistent with the partial order of the BN
- So by the time we sample a node we have values for all of its parents.
- We can then sample from the distribution defined by the CPD and by the chosen values for node's parents.



Analysis of error

University of TEHRAN

- The quality of the estimate obtained depends on the number of particles generated.
- *The Hoeffding bound*
  - guarantees a bound on the absolute error

$$P_{\mathcal{D}}(\hat{P}_{\mathcal{D}}(\boldsymbol{y}) \notin [P(\boldsymbol{y}) - \epsilon, P(\boldsymbol{y}) + \epsilon]) \leq 2e^{-2M\epsilon^2}.$$

  - The number of samples need to bound error by ϵ, with probability at least $1 - \delta$.

$$M \geq \frac{\ln(2/\delta)}{2\epsilon^2}.$$

- The *Chernoff bound*
  - guarantees a bound on the relative error

$$P_{\mathcal{D}}(\hat{P}_{\mathcal{D}}(\boldsymbol{y}) \notin P(\boldsymbol{y})(1 \pm \epsilon)) \leq 2e^{-MP(\boldsymbol{y})\epsilon^2/3}.$$

Conditional probability queries

University of TEHRAN

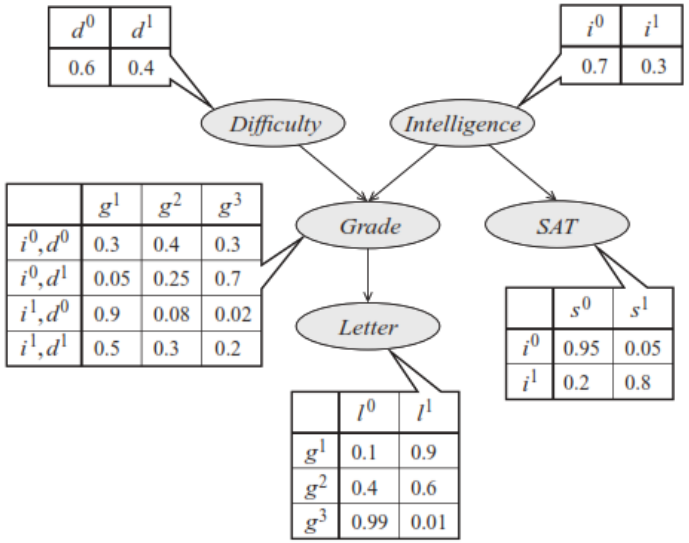
- So far we discuss estimating marginal probabilities
- In general we are interested in conditional probabilities of the form $P(y \mid E = e)$.
  - Much harder
- Rejection sampling
  - we generate samples *x* from the $P(\mathcal{X})$
  - and then reject any sample that is not compatible with *e*
  - the resulting samples are from posterior $P(\mathcal{X} \mid e)$

- Issue with rejection sampling
  - The number of unrejected samples can be quite small (i.e., $MP(e)$).
  - **Example:** if $P(e) = 0.001$ we need to sample $M = 10{,}000$ to on average obtain 10 unrejected particles.

- Low probability of evidence is the rule than the exception.

Forcing the samples to match evidence

- Assume that our evidence is d^1, s^1
  - forward sampling process might generate a value of d^0 for *D*
  - this sample will always be rejected as being incompatible with the evidence.
- A more sensible approach is
  - to simply force the samples to match evidence
  - without correction, this approach can generate incorrect results.
- Example
  - Assume the evidence is s^1
  - Using the naïve process, we first sample D and I, set $S = s^1$, and then sample G and L appropriately.
  - What is the expected fraction of samples that have $I = i^1$?
  - How do you compare it with $\mathrm{P}(I = i^1 \mid s^1)$?



Likelihood weighting

- Likelihood weighting algorithm generated weighted particles according to the likelihood of the evidence (*see section 12.2.1 for some examples and intuition*)

$$\langle \xi[1], w[1] \rangle, \ldots, \langle \xi[M], w[M] \rangle$$

- We then estimate

$$\hat{P}_{\mathcal{D}}(\boldsymbol{y} \mid \boldsymbol{e}) = \frac{\sum_{m=1}^{M} w[m] \mathbf{I}\{\boldsymbol{y}[m] = \boldsymbol{y}\}}{\sum_{m=1}^{M} w[m]}.$$

Algorithm 12.2 Likelihood-weighted particle generation

```
Procedure LW-Sample (
    B,    // Bayesian network over X
    Z = z    // Event in the network
)
Let X_1, ..., X_n be a topological ordering of X
w ← 1
for i = 1, ..., n
    u_i ← x⟨Pa_{X_i}⟩    // Assignment to Pa_{X_i} in x_1, ..., x_{i-1}
    if X_i ∉ Z then
        Sample x_i from P(X_i | u_i)
    else
        x_i ← z⟨X_i⟩    // Assignment to X_i in z
        w ← w · P(x_i | u_i)    // Multiply weight by probability of desired value
return (x_1, ..., x_n), w
```

Importance sampling

University of TEHRAN

- Importance sampling is a general approach for estimating the expectation of a function $f(x)$ relative to some distribution $P(X)$, typically called the *target distribution*.

- We can obtain estimates of this expectation by generating samples from a different distribution Q.

$$\hat{\boldsymbol{E}}_{\mathcal{D}}(f) = \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{x}[m])\frac{P(\boldsymbol{x}[m])}{Q(\boldsymbol{x}[m])}.$$

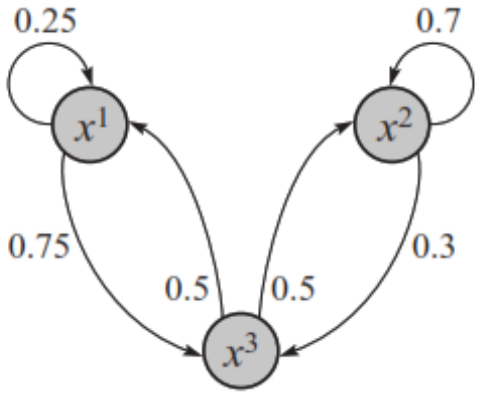
- Reasons to sample from a different distribution:
  - It might be impossible or computationally very expensive to generate samples from P.

University of TEHRAN

Markov Chain Monte Carlo Methods

Markov chain

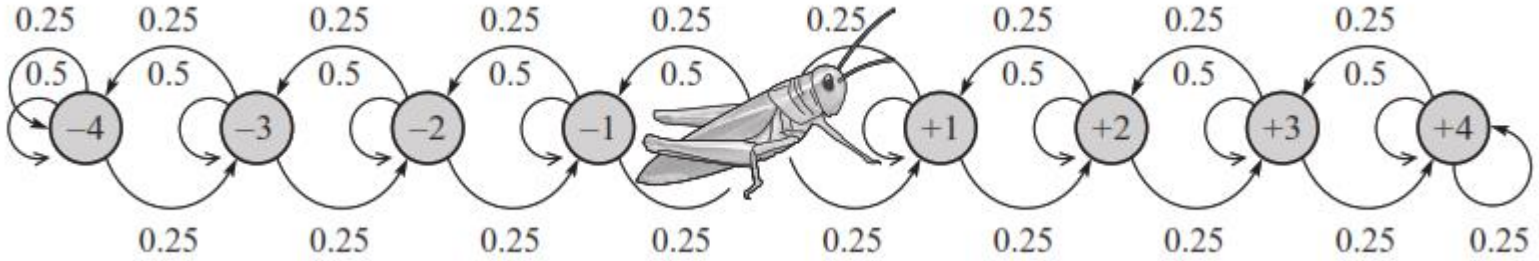
- A Markov chain is defined in terms of a graph of states over which the sampling algorithm takes a random walk
- It defines a probabilistic transition model $\mathcal{T}(x \to x')$ over states x.



- Random sampling process
  - Defines a random sequence of states $x^{(0)}, x^{(1)}, x^{(2)}, \ldots$
  - Because the transition model is random, the state of the process at step *t* can be viewed as a random variable $X^{(t)}$
  - Assume the initial state $X^{(0)}$ is distributed according to some initial state distribution $P^{(0)}(X^{(0)})$

$$P^{(t+1)}(\boldsymbol{X}^{(t+1)} = \boldsymbol{x}') = \sum_{\boldsymbol{x} \in Val(\boldsymbol{X})} P^{(t)}(\boldsymbol{X}^{(t)} = \boldsymbol{x}) \mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}').$$

Example: Grasshopper Markov chain



	-2	-1	0	+1	+2
$\boldsymbol{P^{(0)}}$	0	0	1	0	0
$\boldsymbol{P^{(1)}}$	0	0.25	0.5	0.25	0
$\boldsymbol{P^{(2)}}$			$.5^2 + 2 \times 0.25^2 = 0.375$		
$\boldsymbol{P^{(50)}}$	At t=50, the distribution is almost uniform with a range of 0.1107-0.1116				

The most important aspect of a Markov chain is its long-term behavior

Markov chain Monte carlo (MCMC) sampling

University of TEHRAN

- MCMC sampling is a process that mirrors the dynamics of the Markov chain

Algorithm 12.5 Generating a Markov chain trajectory

Procedure MCMC-Sample (
  $P^{(0)}(\boldsymbol{X})$, // Initial state distribution
  $\mathcal{T}$, // Markov chain transition model
  T // Number of time steps
)

1 Sample $\boldsymbol{x}^{(0)}$ from $P^{(0)}(\boldsymbol{X})$
2 **for** $t = 1, \ldots, T$
3 Sample $\boldsymbol{x}^{(t)}$ from $\mathcal{T}(\boldsymbol{x}^{(t-1)} \rightarrow \boldsymbol{X})$
4 **return** $\boldsymbol{x}^{(0)}, \ldots, \boldsymbol{x}^{(T)}$

- The sample $x^{(t)}$ is drawn from the distribution $P^{(t)}$.
- We are interested in the limit of this process
  - whether $P^{(t)}$ converges
  - and if so, to what limit

Stationary distribution

- Intuitively as the process converges, we would expect $P^{(t+1)}$ to be close to $P^{(t)}$

$$P^{(t)}(\boldsymbol{x}') \approx P^{(t+1)}(\boldsymbol{x}') = \sum_{\boldsymbol{x} \in Val(\boldsymbol{X})} P^{(t)}(\boldsymbol{x}) \mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}').$$

- A distribution $\pi(X)$ is a stationary distribution for a *Markov chain* if it satisfies

$$\pi(\boldsymbol{X} = \boldsymbol{x}') = \sum_{\boldsymbol{x} \in Val(\boldsymbol{X})} \pi(\boldsymbol{X} = \boldsymbol{x}) \mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}').$$

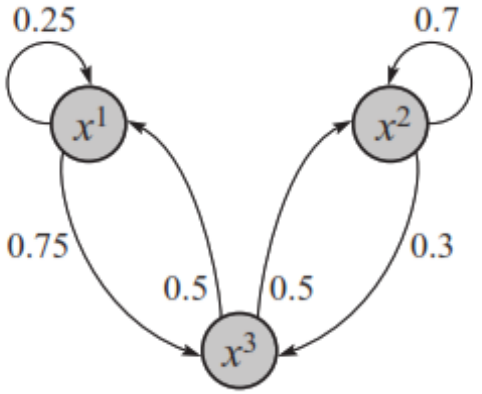
- We already discussed the uniform distribution is a stationary distribution for the Grasshopper Markov chain.

- In general there is no guarantee the stationary distribution is unique.

Stationary distribution-example

- By definition we have the following equations:

$$\begin{aligned}\pi(x^1) &= 0.25\pi(x^1) + 0.5\pi(x^3)\\ \pi(x^2) &= 0.7\pi(x^2) + 0.5\pi(x^3)\\ \pi(x^3) &= 0.75\pi(x^1) + 0.3\pi(x^2),\end{aligned}$$



- in addition, it needs to be a legal distribution:

$$\pi(x^1) + \pi(x^2) + \pi(x^3) = 1.$$

- it is easy to verify the system has a unique solution

$$\begin{aligned}\pi(x^1) &= 0.2\\ \pi(x^2) &= 0.5\\ \pi(x^3) &= 0.3\end{aligned}$$

Regular Markov chains

University of TEHRAN

- A Markov chain is said to be regular if there exists some number *k* such that the probability of getting between every two states in exactly *k* steps is > 0.

- Theorem: if a finite state Markov chain is **regular**, the it has a unique stationary distribution.

- Sufficient conditions for regularity
  - There is a positive probability path between every two states in the state graph.
  - For every state, there is a self-loop.

University of TEHRAN

Using a Markov chain for answering probabilities queries

Using a Markov chain

University of TEHRAN

- The theory of Markov chains provides a general framework for generating samples from a target distribution π.

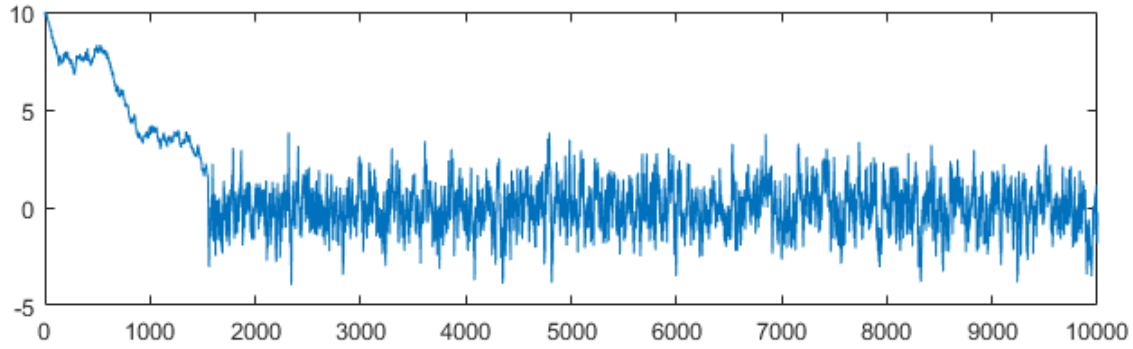
- Goal: compute $P(X \mid E = e)$
  - Too difficult to directly sample from it.
- To this end, we need to build a Markov chain whose unique stationary distribution is $P(X \mid E = e)$
  - We need to define an appropriate transition model

Sample $\boldsymbol{x}^{(0)}$ from $P^{(0)}(\boldsymbol{X})$
for $t = 1, \ldots, T$
  Sample $\boldsymbol{x}^{(t)}$ from $\mathcal{T}(\boldsymbol{x}^{(t-1)} \rightarrow \boldsymbol{X})$

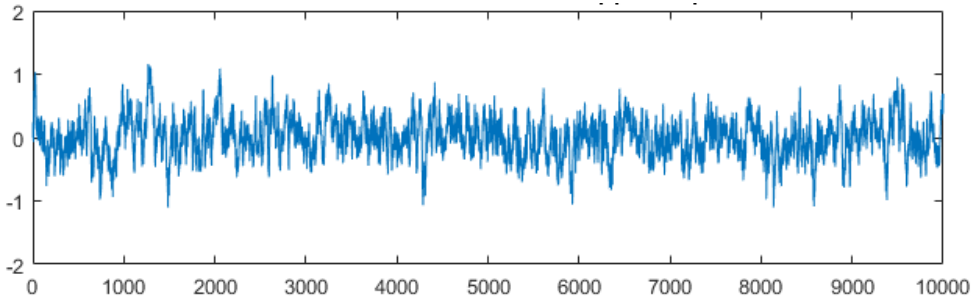
- At early step, $P^{(t)}$ is very different from the P, the stationary distribution

Mixing

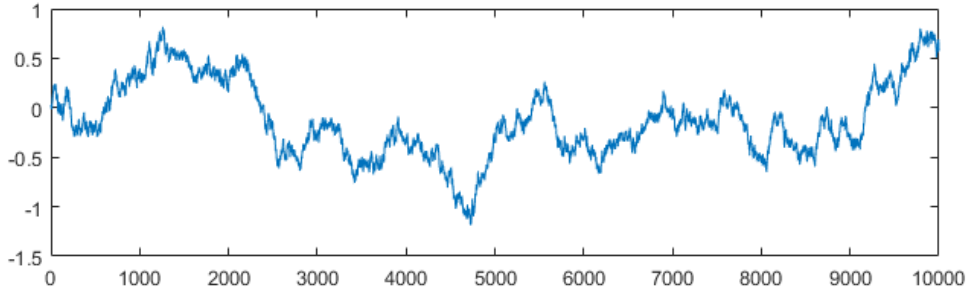
- **The mixing time** of a Markov chain is the time until the Markov chain is "close" to its stationary state distribution.
- *Burn-in time:* the number of steps we take until we collect a sample from the chain.



- How do we evaluate the time required for the a chain to mix?
  - There is no general-purpose theoretical analysis for the mixing time. However,
  - we can compare chain statistics in different windows of a single run
  - or across different runs of the chain



Well-mixed



Highly correlated samples

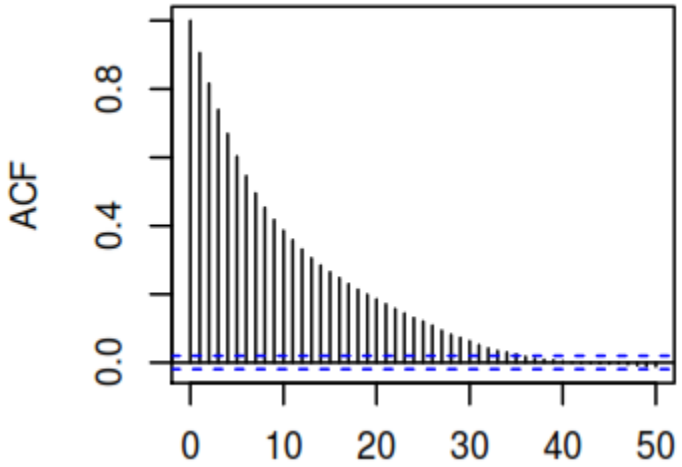
Using the samples

- After discarding the samples in the burn-in phase, we collect M samples from the chain.
- Then we can estimate expectation of any function *f* by

$$\hat{\boldsymbol{E}}_{\mathcal{D}}(f) = \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{x}[m], \boldsymbol{e})$$

- $\hat{E}_D(f)$ is an unbiased estimator $\hat{E}_{\pi(X)}[f(X,e)]$.

- The samples in the chain are correlated
  - If we are looking for a set of independent samples, we only take every *n*th sample
  - Variable *n* is often determined by examining autocorrelation between adjacent samples.



- In many settings, there is no need to look for an independent set of samples
  - Even though correlated, using all samples produces a better estimator

University of TEHRAN

Metropolis-Hastings algorithm

Reversible Markov chains

University of TEHRAN

- Detailed balance equation:

$$\pi(\boldsymbol{x})\mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}') = \pi(\boldsymbol{x}')\mathcal{T}(\boldsymbol{x}' \to \boldsymbol{x}).$$

- **Theorem:** if *T* is regular and it satisfies the detailed balance equation relative to π, then π is the unique stationary distribution of *T*.

- The reversibility condition gives us a condition for verifying that our Markov chain has the desired stationary distribution.

- Metropolis-Hastings algorithm is a general method to build a reversible Markov chain with a particular stationary distribution.
  - Uses the idea of a proposal distribution

Metropolis-Hastings algorithm

University of TEHRAN

- Goal: to design a reversible Markov chain for sampling $\pi(x)$
  - Sampling directly from $\pi(x)$ is hard but it is possible to evaluate it.

- Proposal distribution $\mathrm{T}^{Q}(x \to x')$
- Acceptance probability $\mathrm{A}(x \to x')$
- At each state x, sample x' from $\mathrm{T}^{Q}(x \to x')$
- Accept proposal with probability $\mathrm{A}(x \to x')$
  - If proposal accepted, move to x'.
  - Otherwise stay at x

- The actual transition model

$$\begin{aligned} \mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}') &= \mathcal{T}^{Q}(\boldsymbol{x} \to \boldsymbol{x}')\mathcal{A}(\boldsymbol{x} \to \boldsymbol{x}') \qquad \boldsymbol{x} \neq \boldsymbol{x}' \\ \mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}) &= \mathcal{T}^{Q}(\boldsymbol{x} \to \boldsymbol{x}) + \textstyle\sum_{\boldsymbol{x}' \neq \boldsymbol{x}} \mathcal{T}^{Q}(\boldsymbol{x} \to \boldsymbol{x}')(1 - \mathcal{A}(\boldsymbol{x} \to \boldsymbol{x}')). \end{aligned}$$

Metropolis-Hastings algorithm-2

- Given a proposal distribution, we can use the detailed balance to find $A(x \to x')$ such that the stationary distribution is $\pi(x)$

- The detailed balance equation assert

$$\pi(\boldsymbol{x})\mathcal{T}^Q(\boldsymbol{x} \to \boldsymbol{x}')\mathcal{A}(\boldsymbol{x} \to \boldsymbol{x}') = \pi(\boldsymbol{x}')\mathcal{T}^Q(\boldsymbol{x}' \to \boldsymbol{x})\mathcal{A}(\boldsymbol{x}' \to \boldsymbol{x}).$$

- Hence we obtain

$$\mathcal{A}(\boldsymbol{x} \to \boldsymbol{x}') = \min\left[1, \frac{\pi(\boldsymbol{x}')\mathcal{T}^Q(\boldsymbol{x}' \to \boldsymbol{x})}{\pi(\boldsymbol{x})\mathcal{T}^Q(\boldsymbol{x} \to \boldsymbol{x}')}\right],$$