10707
Deep Learning: Spring 2020

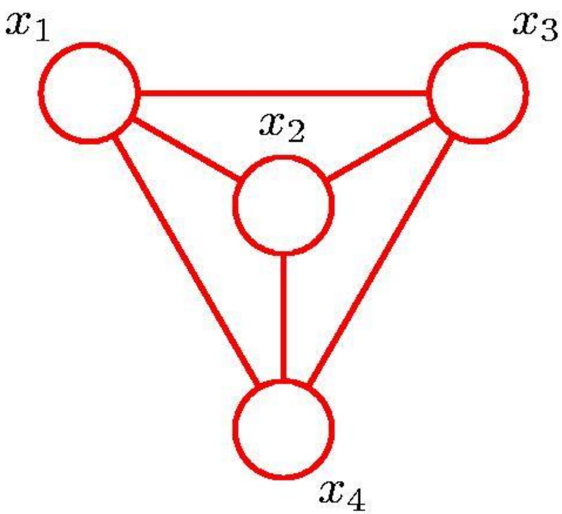
Andrej Risteski

Machine Learning Department

Lecture 12:
Markov Chains, discrete and continuous

Graphical Models

Recall: graph contains a set of nodes connected by edges.



In a probabilistic graphical model, each node represents a random variable, links represent "probabilistic dependencies" between random variables.

Graph specifies how joint distribution over all random variables decomposes into a **product** of factors, each factor depending on a subset of the variables.

Two types of graphical models:

- **Bayesian networks**, also known as Directed Graphical Models (the links have a particular directionality indicated by the arrows)
- **Markov Random Fields**, also known as Undirected Graphical Models (the links do not carry arrows and have no directional significance).

Algorithmic pros/cons of latent-variable models (so far)

RBM's

- Hard to draw samples

(In fact, #P-hard provably, even in Ising models)

- Easy to sample posterior distribution over latents

Directed models

- Easy to draw samples

- Hard to sample posterior distribution over latents

(In fact, #P-hard even in mixtures)

Canonical tasks with graphical models

Inference

Given values for the parameters θ of the model, *sample/calculate* marginals (e.g. sample $p_\theta(x_1), p_\theta(x_4, x_5), p_\theta(z|x)$, etc.)

Learning

Find values for the parameters θ of the model, that give a *high likelihood* for the observed data. (e.g. canonical way is solving maximum likelihood optimization

$$\max_{\theta \in \Theta} \sum_{i=1}^{n} \log p(x_i)$$

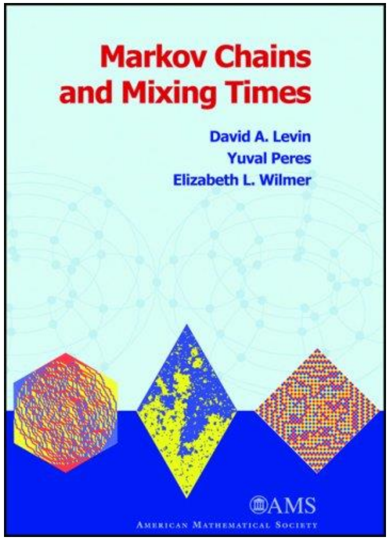
Other methods exist, e.g. method of moments (matching moments of model), but less used in deep learning practice.

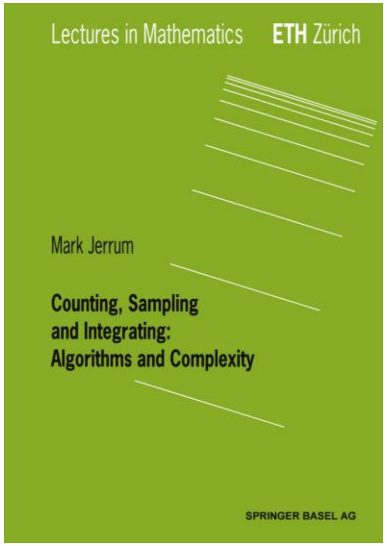
Algorithmic approaches

When faced with a difficult to calculate probabilistic quantity (partition function, difficult posterior), there are two families of approaches:

MARKOV CHAIN MONTE CARLO

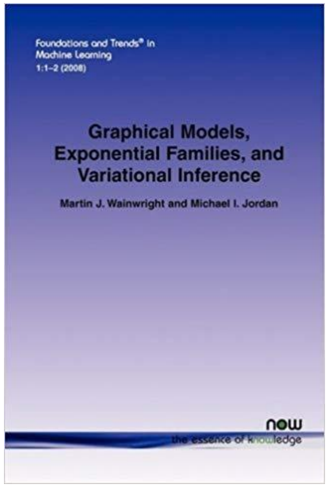
- Random walk w/ equilibrium distribution the one we are trying to sample from.
- Well studied in TCS.





VARIATIONAL METHODS

- Based on solving an optimization problem.
- Very popular in practice.
- Comparatively poorly understood



Goal for the day:

Sample from distributions given up to a constant of proportionality

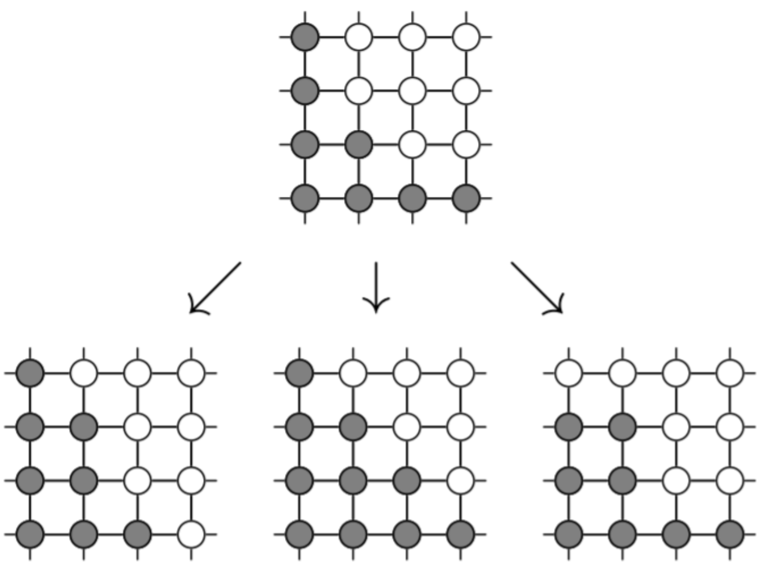
$$P_\theta(\mathbf{x}) = \frac{1}{\mathcal{Z}(\theta)} \exp\Big(\sum_{ij \in E} x_i x_j \theta_{ij} + \sum_{i \in V} x_i \theta_i \Big) \qquad p(z|x) = \frac{p(z)p(x|z)}{p(x)}$$

Part I: Common random walks over discrete domains

Sampling via random walks

Goal: Sample from distribution given up to constant of proportionality.

Idea: explore domain via *random, local* moves



Hope: enough moves $\Rightarrow$ the random process "forgets" starting point, follows the distr. we are trying to sample.

Sampling via random walks

Goal: Sample from distribution given up to constant of proportionality.

Definition: A set of random variables $(X_1, X_2, \dots, X_T)$ is **Markov** if $\forall t: P(X_t | X_{<t}) = P(X_t | X_{t-1})$

It is homogeneous if $P(X_t | X_{t-1})$ doesn't depend on t.

We can describe a homogeneous Markov process on a discrete domain $\mathcal{X}$ by a **transition matrix** $T \in \mathbb{R}_+^{|\mathcal{X}| \times |\mathcal{X}|}$: $T_{ij} = P(X_{t+1} = j \,| X_t = i)$

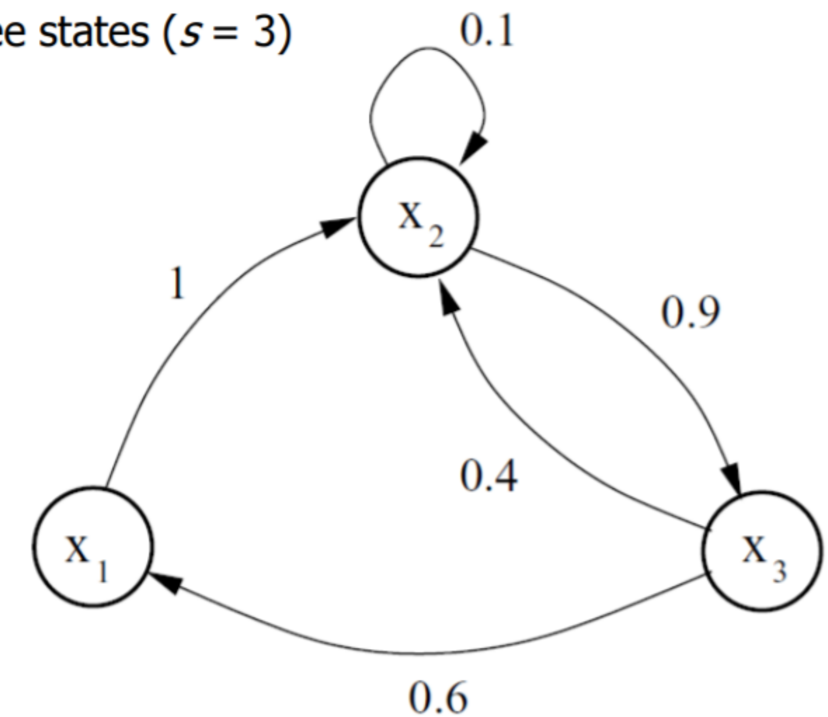
Clearly, $\forall i, \sum_j \; T_{ij} = 1$. We will also call such process a Markov Chain/ Markov random walk.

Example

Markov chain with three states ($s = 3$)

$$T = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0.1 & 0.9 \\ 0.6 & 0.4 & 0 \end{bmatrix}$$

Transition matrix



Transition graph

Stationary distribution

Stationary distribution: a distribution $\pi = (\pi_1, \dots \pi_{|\mathcal{X}|})$ is stationary for a Markov walk if $\pi T = \pi$.

In other words: if we start with a sample of π and transition according to T, we end with a sample following π as well.

$$(0.22, 0.41, 0.37) \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0.1 & 0.9 \\ 0.6 & 0.4 & 0 \end{bmatrix} = (0.22, 0.41, 0.37)$$

Stationary distribution need not be unique: e.g. T is the identity matrix.

Many Markov Chains have unique stationary distributions: after taking many steps, starting with any distribution, we get to the same distribution

$$\forall p_0, \lim_{t \to \infty} p_0 T^t = \pi$$

In other words, eventually, the chain "forgets" the starting point.

Stationary distribution

Stationary distribution: a distribution $\pi = (\pi_1, \dots \pi_{|\chi|})$ is stationary for a Markov walk if $\pi T = \pi$.

Many Markov Chains have unique stationary distributions: after taking many steps, starting with any distribution, we get to the same distribution

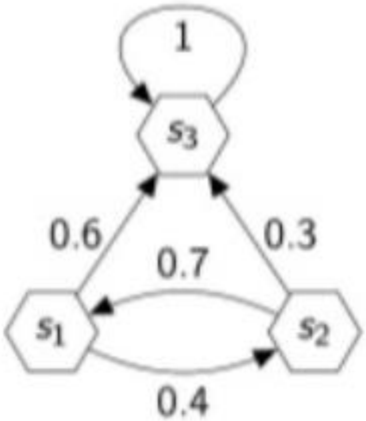
$$\forall p_0, \lim_{t \to \infty} p_0 T^t = \pi$$

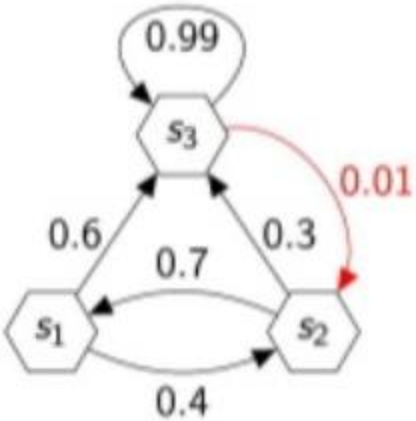
Name of the game: if we wish to sample from some π, design a Markov Chain which has π as stationary distribution.

If we run chain long enough (??), we can draw samples from something close to π

Conditions for having a unique stationary distribution

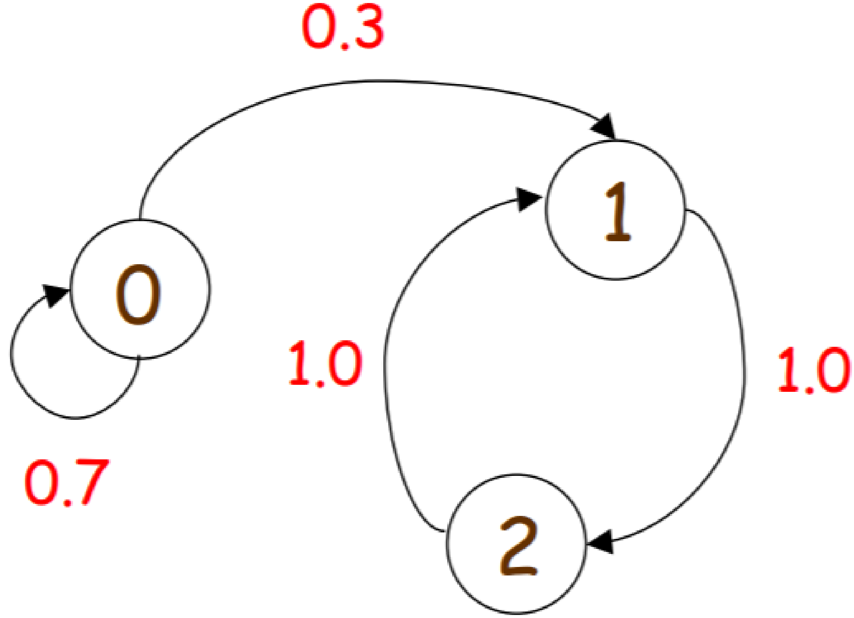
Potential problem: transition graph is not connected.





Conditions for having a unique stationary distribution

Potential problem: there are cycles in graph



Conditions for having a unique stationary distribution

These are all the possible problems!

Irreducibility: there is a path that transitions from any state to any other.
For each pairs of states (i,j), there is a positive probability, starting in state i, that the process will ever enter state j.

= Transition graph is connected;

Aperiodicity: random walk doesn't get trapped in cycles.
A state i is aperiodic if there exists n s.t., $\forall n' \geq n, P\left(X_{n'} = i \middle| X_0 = i\right) > 0$.
If all states are aperiodic, chain is called aperiodic.

Thm: for any *irreducible+aperiodic* Markov chain there is a unique π, s.t.

$$\forall p_0, \lim_{t \to \infty} p_0 T^t = \pi$$

Detailed balance

Useful sufficient condition for π to be a stationary distribution: detailed balance.

$$\pi_i T_{ij} = \pi_j T_{ji}, \forall (i, j)$$

Proof:

$$(\pi T)_i = \sum_j \pi_j T_{ji} = \sum_j \pi_i T_{ij}$$

$$= \pi_i \sum_j T_{ij}$$

$$= \pi_i$$

Metropolis-Hastings

Suppose we are trying to sample from π defined over a domain of size m (think m is very large, like in Ising models), up to a constant of proportionality:

$$\pi(x = i) = \frac{b(i)}{Z}, Z = \sum_{i=1}^{m} b(i)$$

Metropolis-Hastings: random walk assuming an "easy-to-sample from" transition kernel q(i,j), along with "corrections".

Metropolis-Hastings

Suppose we have an easy to sample from "transition kernel" q(i,j).

Consider the following random walk, for some $\alpha(i,j)$ we will pick:

$$\Pr(X_n = j \,|X_{n-1} = i) =$$

1., from state i go to state j with prob. $q(i,j)$

2., $\begin{cases} \text{with prob } 1-\alpha(i,j) \text{ go back to state } i, \\ \text{with prob } \alpha(i,j) \text{ stay in state } j. \end{cases}$

Then, we have:

$$\begin{aligned} P(X_{n+1} = j|X_n = i) &= q(i,j)\alpha(i,j) \quad \forall j \neq i \\ P(X_{n+1} = i|X_n = i) &= q(i,i) + \sum_{k \neq i} q(i,k)(1-\alpha(i,k)) \end{aligned}$$

Metropolis-Hastings

Observation

$$\pi_i P_{ij} = \pi_j P_{ji} \quad \forall j \neq i \Leftrightarrow \pi_i q(i,j)\alpha(i,j) = \pi_j q(j,i)\alpha(j,i) \quad \forall j \neq i \quad (*)$$

Proof: $P_{ij} = P(X_{n+1} = j | X_n = i) = q(i,j)\alpha(i,j) \ \forall j \neq i$

Theorem

$$\text{If } \alpha(i,j) \ = \ \min\left(\frac{\pi_j q(j,i)}{\pi_i q(i,j)}, 1\right) = \min\left(\frac{b(j)q(j,i)}{b(i)q(i,j)}, 1\right)$$

$$\Rightarrow \ (\pi_1, \ldots \pi_m) \text{ stationary distribution}$$

Note, this only depends on unnormalized distribution (b(i) values)

Proof:

$$\text{If } \alpha(i,j) = \frac{\pi_j q(j,i)}{\pi_i q(i,j)} \Leftrightarrow \alpha(j,i) = 1$$

=> Detailed balance (*) holds

Gibbs sampling

Consider sampling a distribution over n variables $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$, s.t. each of the conditional distributions $P(x_i|\boldsymbol{x}_{-i})$ is easy to sample. :

e.g. recall Ising models: $P_\theta(x_i = 1|\mathbf{x}_{-i}) = \dfrac{1}{1 + \exp(-\theta_i - \sum_{ij \in E} x_j \theta_{ij})}$,

A common way to do this is using **Gibbs sampling**:

Repeat:

Let current state be $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$

Pick $i \in [n]$ uniformly at random.

Sample x $\sim P(X_i = x|\boldsymbol{x}_{-i})$

Update state to $\boldsymbol{y} = (x_1, x_2, \dots, x_{i-1}, x, x_{i+1}, \dots, x_n)$

Gibbs sampling

Repeat:

Let current state be $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$

Pick $i \in [n]$ uniformly at random.

Sample x $\sim P(X_i = x | \boldsymbol{x}_{-i})$

Update state to $\boldsymbol{y} = (x_1, x_2, \dots, x_{i-1}, x, x_{i+1}, \dots, x_n)$

Why does it work? Metropolis-Hastings with appropriate kernel!

Let

$$\begin{aligned} q(\mathbf{x}, \mathbf{y}) &= q(\overbrace{(x_1, \dots, x_n)}^{\mathbf{x}}, \overbrace{(x_1, \dots, x_{i-1}, x, x_{i+1}, x_n)}^{\mathbf{y}}) \\ &\doteq \frac{1}{n} P(X_i = x | X_j = x_j, \forall j \neq i) \\ &= \frac{1}{n} \frac{P(\mathbf{y})}{P(X_j = x_j, \forall j \neq i)} \end{aligned}$$

Gibbs sampling

Why does it work? Metropolis-Hastings with appropriate kernel!

Let

$$
\begin{aligned}
q(\mathbf{x},\mathbf{y}) &= q(\overbrace{(x_1,\ldots,x_n)}^{\mathbf{x}},\overbrace{(x_1,\ldots,x_{i-1},x,x_{i+1},x_n)}^{\mathbf{y}}) \\
&\doteq \frac{1}{n}P(X_i=x|X_j=x_j,\forall j\neq i) \\
&= \frac{1}{n}\frac{P(\mathbf{y})}{P(X_j=x_j,\forall j\neq i)}
\end{aligned}
$$

Shouldn't we reject occasionally? **No**:

Theorem

If $\alpha(i,j) \;=\; \min\left(\frac{\pi_j q(j,i)}{\pi_i q(i,j)},1\right)=\min\left(\frac{b(j)q(j,i)}{b(i)q(i,j)},1\right)$

$\Rightarrow \;(\pi_1,\ldots\pi_m)$ stationary distribution

$$
\frac{p(\mathbf{y})q(\mathbf{y},\mathbf{x})}{p(\mathbf{x})q(\mathbf{x},\mathbf{y})}=\frac{p(\mathbf{y})\frac{1}{n}\frac{P(\mathbf{x})}{P(Y_j=y_j,j\neq i)}}{p(\mathbf{x})\frac{1}{n}\frac{P(\mathbf{y})}{P(X_j=x_j,j\neq i)}}
$$

Gibbs sampling

Why does it work? Metropolis-Hastings with appropriate kernel!

Let

$$\begin{aligned} q(\mathbf{x}, \mathbf{y}) &= q(\overbrace{(x_1, \ldots, x_n)}^{\mathbf{x}}, \overbrace{(x_1, \ldots, x_{i-1}, x, x_{i+1}, x_n)}^{\mathbf{y}}) \\ &\doteq \frac{1}{n} P(X_i = x | X_j = x_j, \forall j \neq i) \\ &= \frac{1}{n} \frac{P(\mathbf{y})}{P(X_j = x_j, \forall j \neq i)} \end{aligned}$$

Shouldn't we reject occasionally? **No**:

$$\frac{p(\mathbf{y}) q(\mathbf{y}, \mathbf{x})}{p(\mathbf{x}) q(\mathbf{x}, \mathbf{y})} = \frac{p(\mathbf{y}) \frac{1}{n} \frac{P(\mathbf{x})}{P(Y_j = y_j, j \neq i)}}{p(\mathbf{x}) \frac{1}{n} \frac{P(\mathbf{y})}{P(X_j = x_j, j \neq i)}} = \frac{p(\mathbf{y}) p(\mathbf{x})}{p(\mathbf{x}) p(\mathbf{y})} = 1$$

since $P(X_j = x_j, j \neq i) = P(Y_j = y_j, j \neq i)$

What governs "mixing time"

So far, we've only worried about designing chains s.t. $\forall p_0, \lim_{t\to\infty} p_0 T^t = \pi$

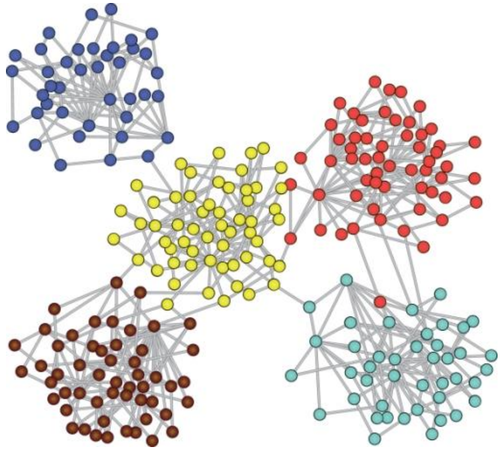
But, we're running this in practice, so want for sensible t, $\forall p_0,\ p_0 T^t \approx \pi$

(Appropriately formalized, this is called *mixing time*.)

There is no silver bullet for analyzing general transition T, but one common tool is *conductance*: which essentially says the transition graph doesn't have "bottlenecks".

The conductance of a subset S is defined as:

$$\phi(S) = \frac{\sum_{i\in S, j\notin S} T_{ij}}{\sum_{i\in S} \pi_i}$$



(e.g. how easy it is to leave S, given that we started in S)

(e.g. the colored sets have poor conductance)

What governs "mixing time"

So far, we've only worried about designing chains s.t. $\forall p_0, \lim_{t\to\infty} p_0 T^t = \pi$

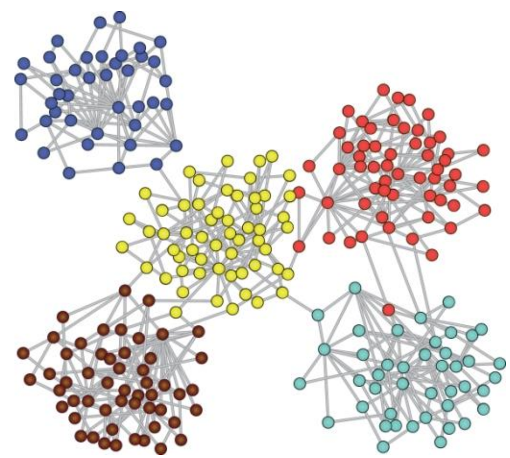
But, we're running this in practice, so want for sensible t, $\forall p_0,\ p_0 T^t \approx \pi$

(Appropriately formalized, this is called *mixing time*.)

There is no silver bullet for analyzing general transition T, but one common tool is *conductance*: which essentially says the transition graph doesn't have "bottlenecks".

The conductance of a subset S is defined as:

$$\phi(S) = \frac{\sum_{i\in S, j\notin S} T_{ij}}{\sum_{i\in S} \pi_i}$$



It's clear that sets of poor $\phi(S)$ impede mixing time:

If we start at S, even with the correct π, it'll take us long to leave S.

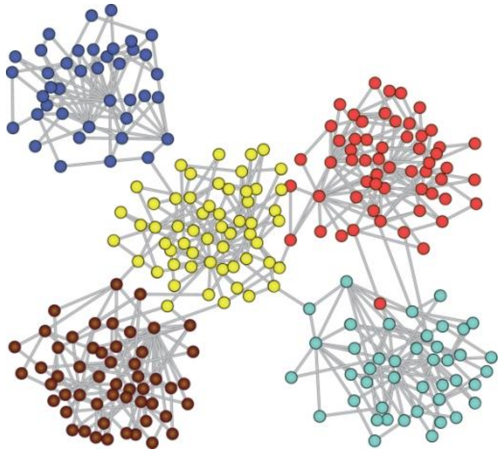
What governs "mixing time"

So far, we've only worried about designing chains s.t. $\forall p_0, \lim_{t\to\infty} p_0 T^t = \pi$

But, we're running this in practice, so want for sensible t, $\forall p_0,\ p_0 T^t \approx \pi$

(Appropriately formalized, this is called *mixing time*.)

There is no silver bullet for analyzing general transition T, but one common tool is *conductance*: which essentially says the transition graph doesn't have "bottlenecks".



The conductance of a subset S is defined as:

$$\phi(S) = \frac{\sum_{i\in S, j\notin S} T_{ij}}{\sum_{i\in S} \pi_i}$$

It's clear that sets of poor $\phi(S)$ impede mixing time:

The distribution is "multimodal": has S's that have large probability, but are difficult to transition between.

What governs "mixing time"

So far, we've only worried about designing chains s.t. $\forall p_0, \lim_{t\to\infty} p_0 T^t = \pi$

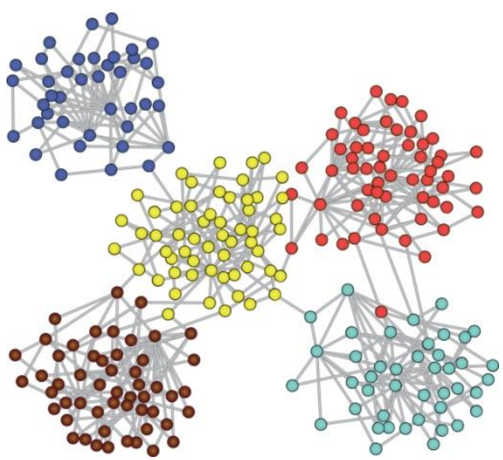
But, we're running this in practice, so want for sensible t, $\forall p_0,\ p_0 T^t \approx \pi$

(Appropriately formalized, this is called *mixing time*.)

There is no silver bullet for analyzing general transition T, but one common tool is *conductance*: which essentially says the transition graph doesn't have "bottlenecks".

The conductance of a subset S is defined as:

$$\phi(S) = \frac{\sum_{i\in S, j\notin S} T_{ij}}{\sum_{i\in S} \pi_i}$$



Conversely, if $\phi(S)$ is large for all S => mixing time is good!

What governs “mixing time”

So far, we’ve only worried about designing chains s.t. $\forall p_0, \lim_{t\to\infty} p_0 T^t = \pi$

But, we’re running this in practice, so want for sensible t, $\forall p_0,\ p_0 T^t \approx \pi$

(Appropriately formalized, this is called *mixing time*.)

Note common misconception: random walk must visit each state in domain to mix.

This is of course not true! (There does however need to be a reasonable **probability** that some set of moves gets us anywhere in the domain.)

(Otherwise, what would be the point of running a Markov Chain as opposed to brute force calculation of the partition function…)

Part II: Random walks over continuous domains (*Langevin dynamics*)

Langevin dynamics

Consider sampling from $p(x) = \frac{1}{Z}\exp(-f(x))$ with support $\mathbb{R}^d$, f(x) is differentiable and we can efficiently take gradients. (e.g. f(x) is parametrized by a neural network).

A natural random walk:

Limit (as $\eta \to 0$) of: $$x_{t+1} = \overbrace{x_t - \eta \nabla f(x_t)}^{\text{Gradient descent}} + \overbrace{\sqrt{2\eta}\xi_k}^{\text{Gaussian noise}}$$

$$\xi_k \sim N(0, I)$$

Stationary (equilibrium) distr.

$$p(x) = \frac{1}{Z}\exp(-f(x))$$

The dichotomy

$$p(x) \propto e^{-f(x)}$$

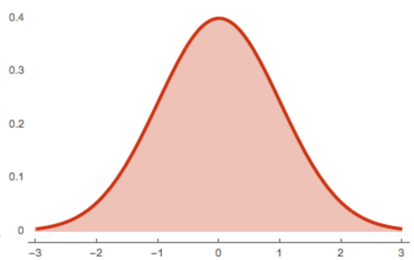
Log-concave distribution (f convex)

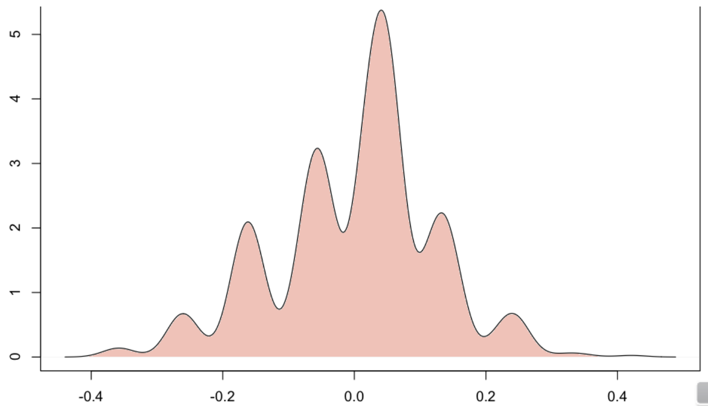
- Provable algorithms, but practically restrictive (unimodal)

Non-log-concave distributions

- #P-hard in the worst-case
- Common source of hardness: multimodality

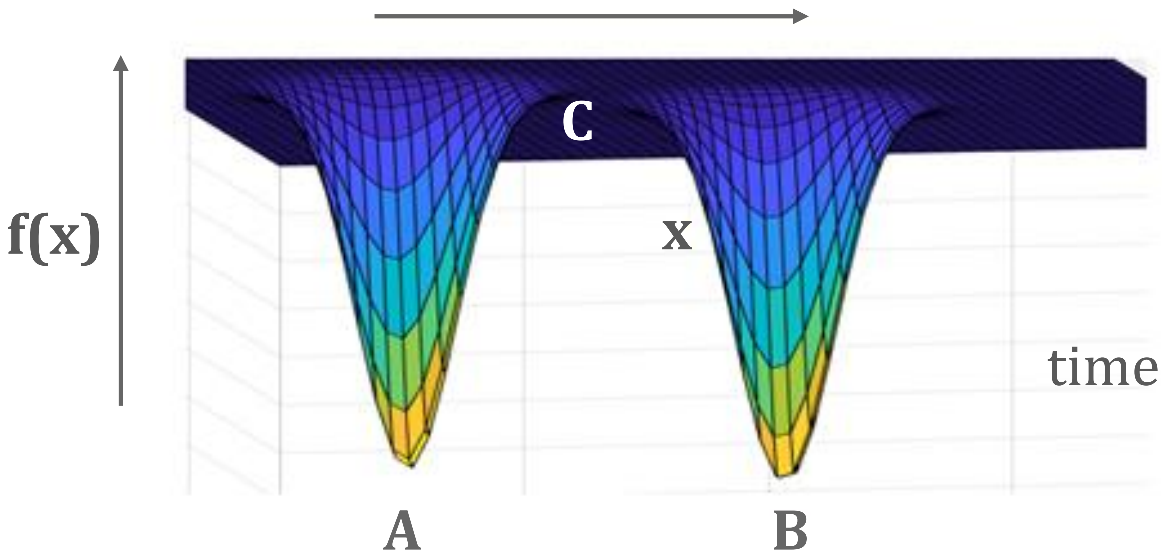
<u>Parallel to optimization</u>: if f is convex, minimizing f is easy.
(All local minima are global ⇒ gradient descent works.)





Why multimodality is trouble

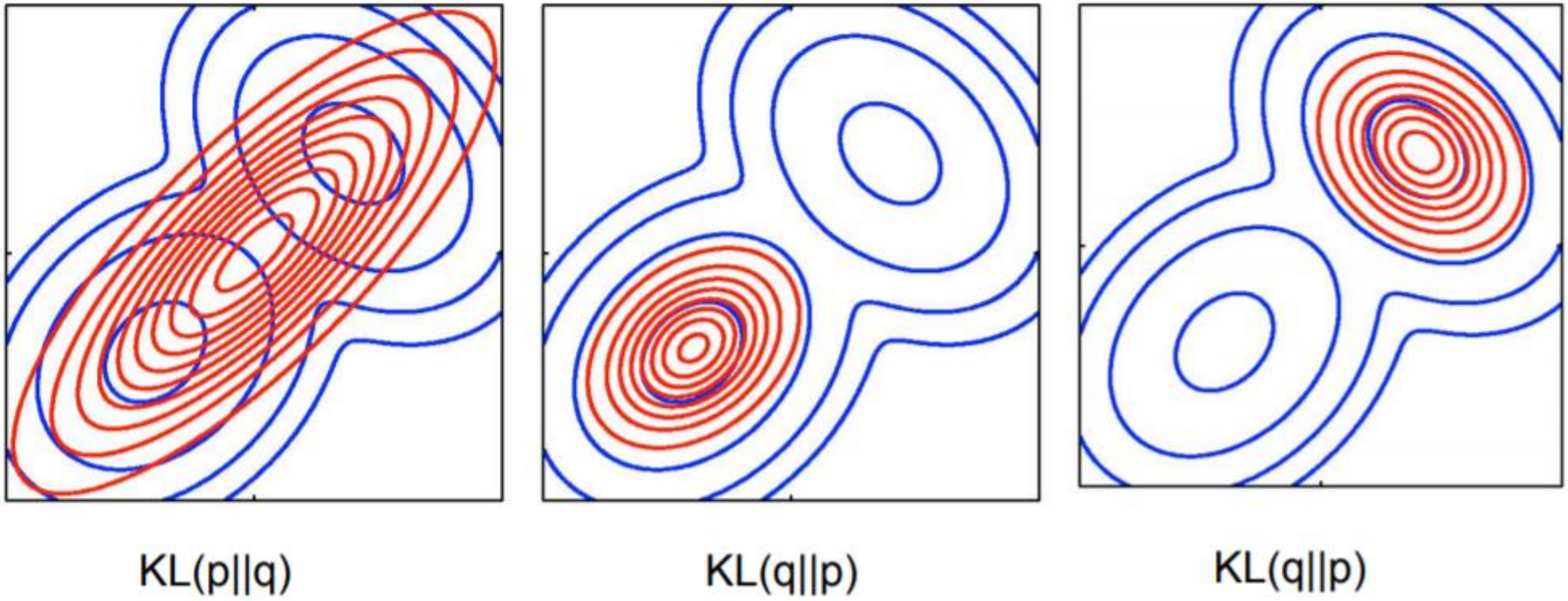
__Sharp hills are hard to climb!__



Recall, peaks of p = valleys of f

(Bovier '02,'04) ⇒
time to get from A to B (through C) can be exponential!

Same problem we had with variational methods!!

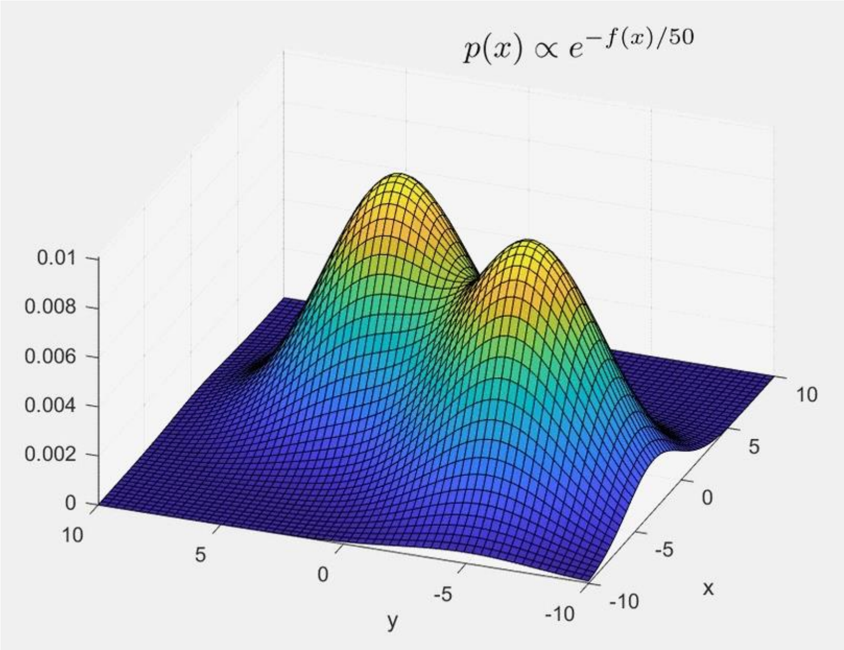


Potential solutions for multimodality

Unlike optimization,
scale (**temperature**) matters!

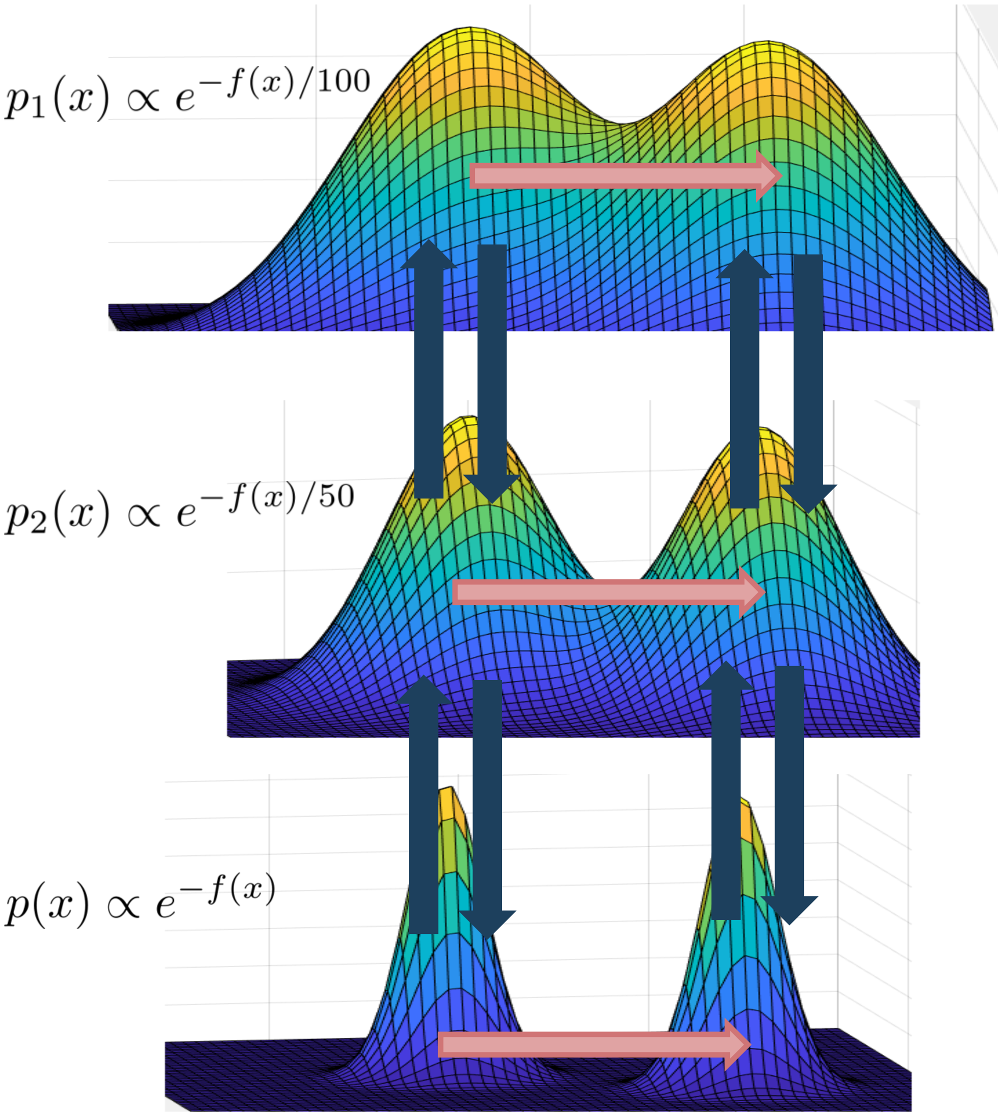
Sampling flatter distributions is easier!

Can we leverage this?



Tempering/annealing: run multiple chains at different temperatures, and use the fact that chains at higher temperatures move faster through landscape.

Tempering: flatten the hills



Algorithm: run multiple walks in parallel for **different temperatures**. **Swap** locations **occasionally** so lower-temp. chains explore space faster. (Occasionally = equilibrium distr. at each temperature is correct.)

Popular in practice, among other "annealing tricks" ((reverse) annealed importance sampling, tunneling…).

Little theory…

The tempering chain

- Markov Chain on state space: $\mathbb{R}^d \times [L]$. (L is # of temperatures)
- Let M_k be the Markov Chain corresponding to temperature k (i.e. with stationary distr. p_k)

<u>Tempering chain.</u>

Run k-th chain

Let current point be (x, k).

- With probability ½

Set next point

Swap points, perform
Metropolis-Hastings to
preserve stationary distr.

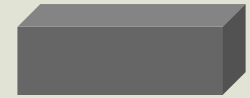
- With probability ½: pick … set next point to (x,k') with probability $\min\left(\frac{p_{k'}(x)}{p_k(x)}, 1\right)$

The stationary distribution is $P(x,t) = \frac{1}{L} p_k(x)$

When does this work?

Each component a “**mode**”
“**Modes**” have same shape

Unknown means!
(**gradient** access)

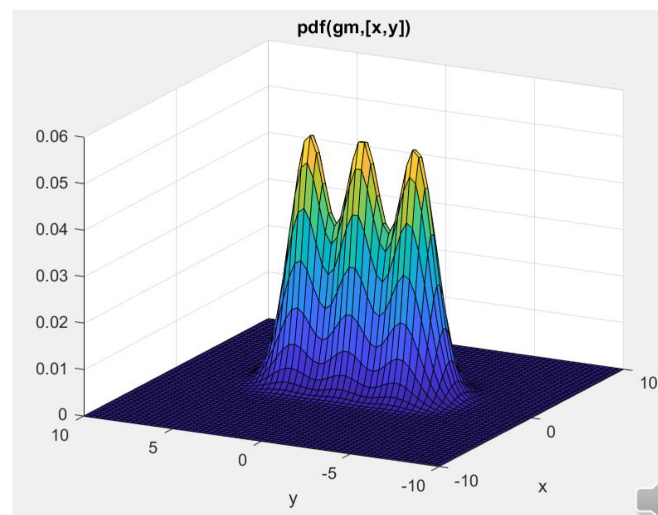


Thm [Ge-Lee-Risteski ‘18**]:**

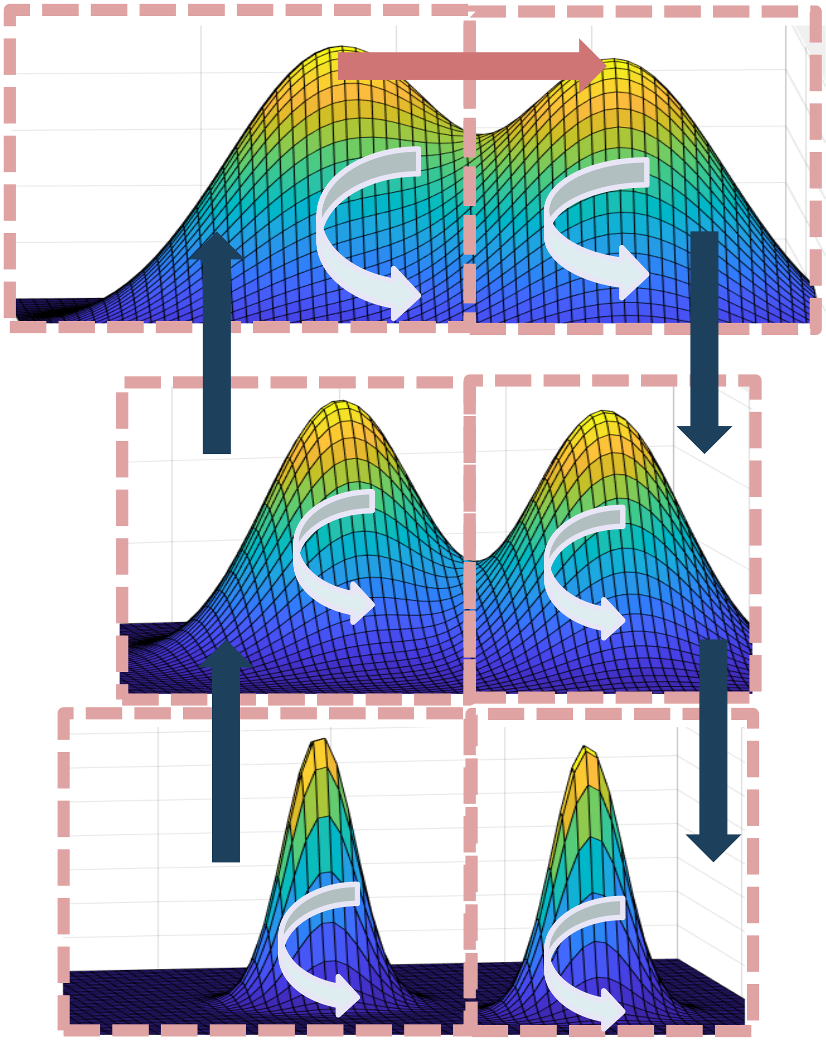
Let $p(x) \propto e^{-f(x)}$ be a **mixture** of **n shifts of a d-dim. log-concave distribution.**

Then**, Langevin + simulated tempering** run for time **poly(n, d)** samples from distribution close to **p**

Result is **robust**: works if p is **close** to a mixture (w/ degradation in runtime)



Why it works: take the road less hilly



Choose **highest temp.** s.t. walk converges fast. (Hills are flat)

Can partition space in **blocks** (≈ modes), s.t.

(1) Walk converges fast **inside each block**

(2) Blocks **aren't too small**

⇒ Fast convergence for tempering.

Intuition: Fast inside each mode.

Can get to highest temp. "parallel" mode.

Can get to any other mode at highest temp.

Can get to lowest temp. "parallel" mode.