Learning and sampling multimodal distributions with data-based initialization

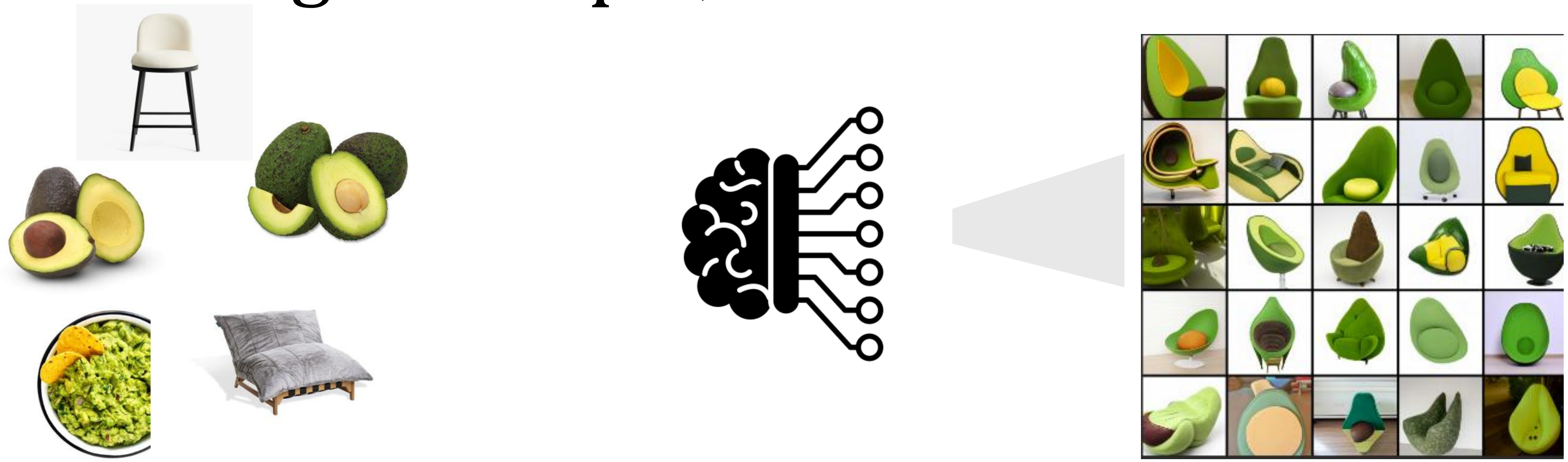
Thuy-Duong “June” Vuong

UC Berkeley

Joint work with Holden Lee and Frederic Koehler

Learning to sample, a central task in GenAI

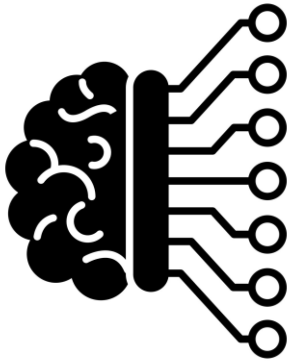
Learning to sample, a central task in GenAI

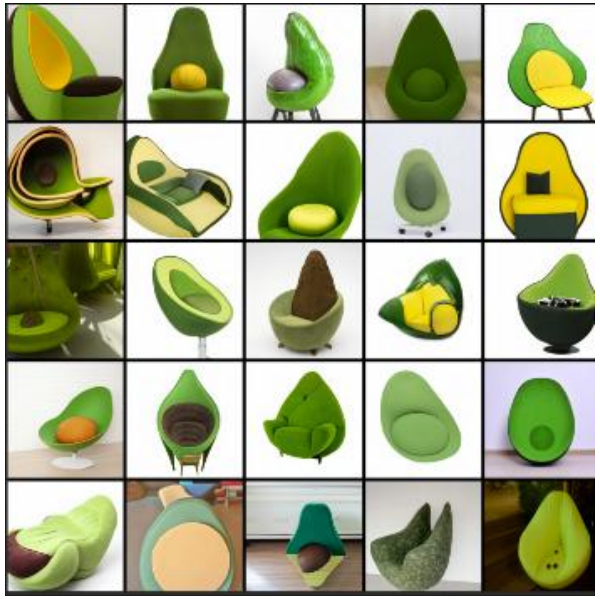


Training samples

Generated samples

Learning to sample, a central task in GenAI





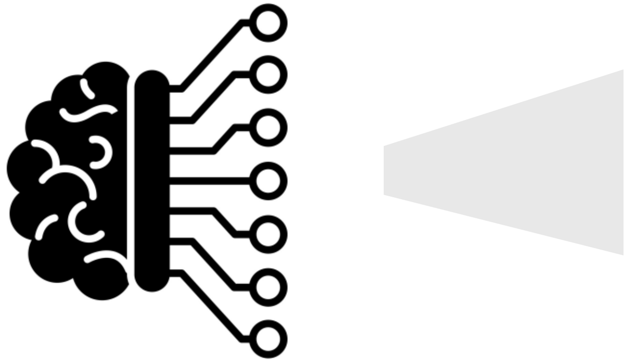
Training samples

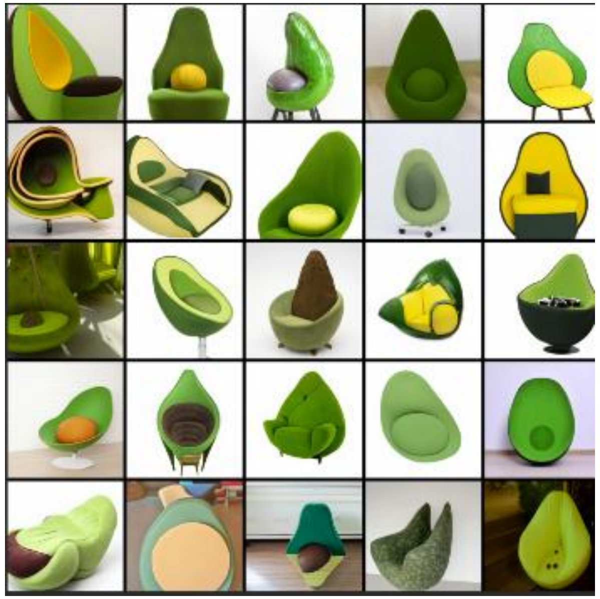
$y_1, \cdots, y_n \sim \pi$

Generated samples

$\mathcal{A}(\underbrace{r}_{\text{Random source}}; (y_i)) \to y'$

Learning to sample, a central task in GenAI





Training samples

$y_1, \cdots, y_n \sim \pi$

$\hat{\pi} \equiv \hat{\pi}_{(y_i)_{i=1}^n}$

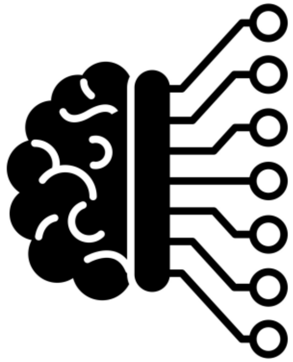
$= Dist(\mathcal{A}(r; (y_i)) | (y_i))$

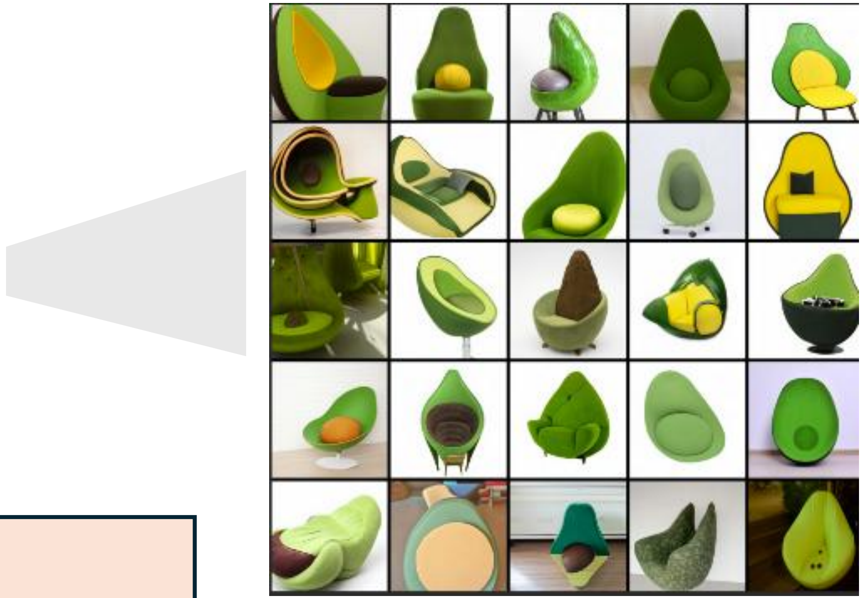
Generated samples

$\mathcal{A}(r; (y_i)) \to y'$

$d_{TV}\left(\hat{\pi}_{(y_i)_{i=1}^n}, \pi\right) \leq \epsilon$

Learning to sample, a central task in GenAI





Training samples

$y_1, \cdots, y_n \sim \pi$

$\hat{\pi} \equiv \hat{\pi}_{(y_i)_{i=1}^n}$
$= Dist(\mathcal{A}(r; (y_i)) | (y_i))$

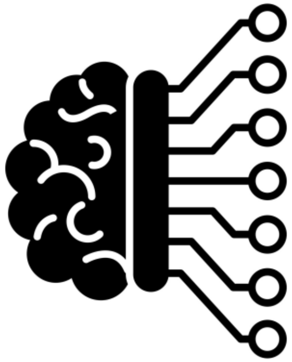
Impossible for atypical $y_1, \cdots, y_n$!

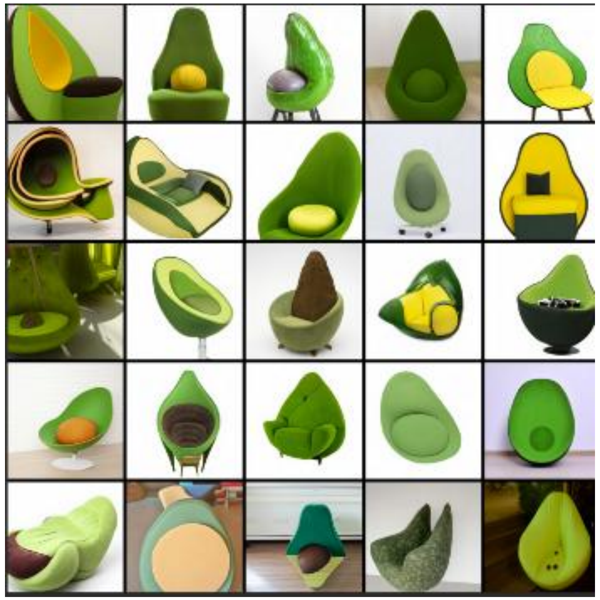
Generated samples

$\mathcal{A}(r; (y_i)) \to y'$

$d_{TV}\left(\hat{\pi}_{(y_i)_{i=1}^n}, \pi\right) \le \epsilon$

Learning to sample, a central task in GenAI





Training samples

$y_1, \cdots, y_n \sim \pi$

$\hat{\pi} \equiv \hat{\pi}_{(y_i)_{i=1}^n}$
$= Dist(\mathcal{A}(r; (y_i)) | (y_i))$

W. prob $\geq 1 - \delta$ over
$(y_i)_{i=1}^n \sim \pi$

Generated samples

$\mathcal{A}(r; (y_i)) \rightarrow y'$

$d_{TV}\left(\hat{\pi}_{(y_i)_{i=1}^n}, \pi\right) \leq \epsilon$

Learning to sample

- How to construct $\mathcal{A}$?
- Use **random walk**

- Input: Training samples $y_1, \cdots, y_n \sim \pi$ i.i.d.
- Output: Algorithm $\mathcal{A}$ that generates many new samples
- Guarantee:

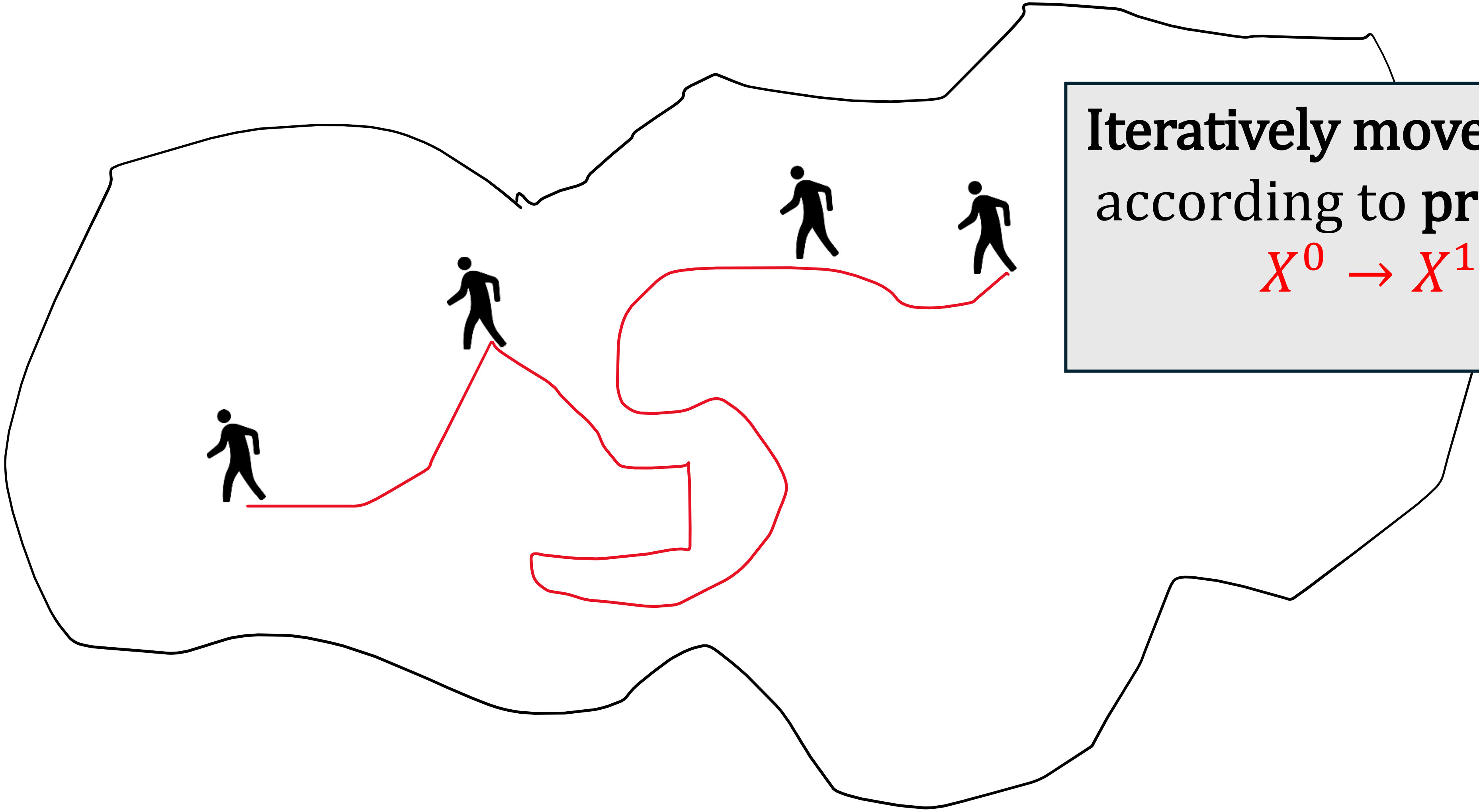
Let $\hat{\pi}_Y = Dist(\mathcal{A}(u, (y_i)_{i=1}^n) | Y = \{y_1, \ldots, y_n\})$

With probability $\geq 1 - \delta$ over $y_1, \cdots, y_n \sim \pi$,

$$d_{TV}(\hat{\pi}_Y, \mu) \leq \epsilon$$

Algorithm

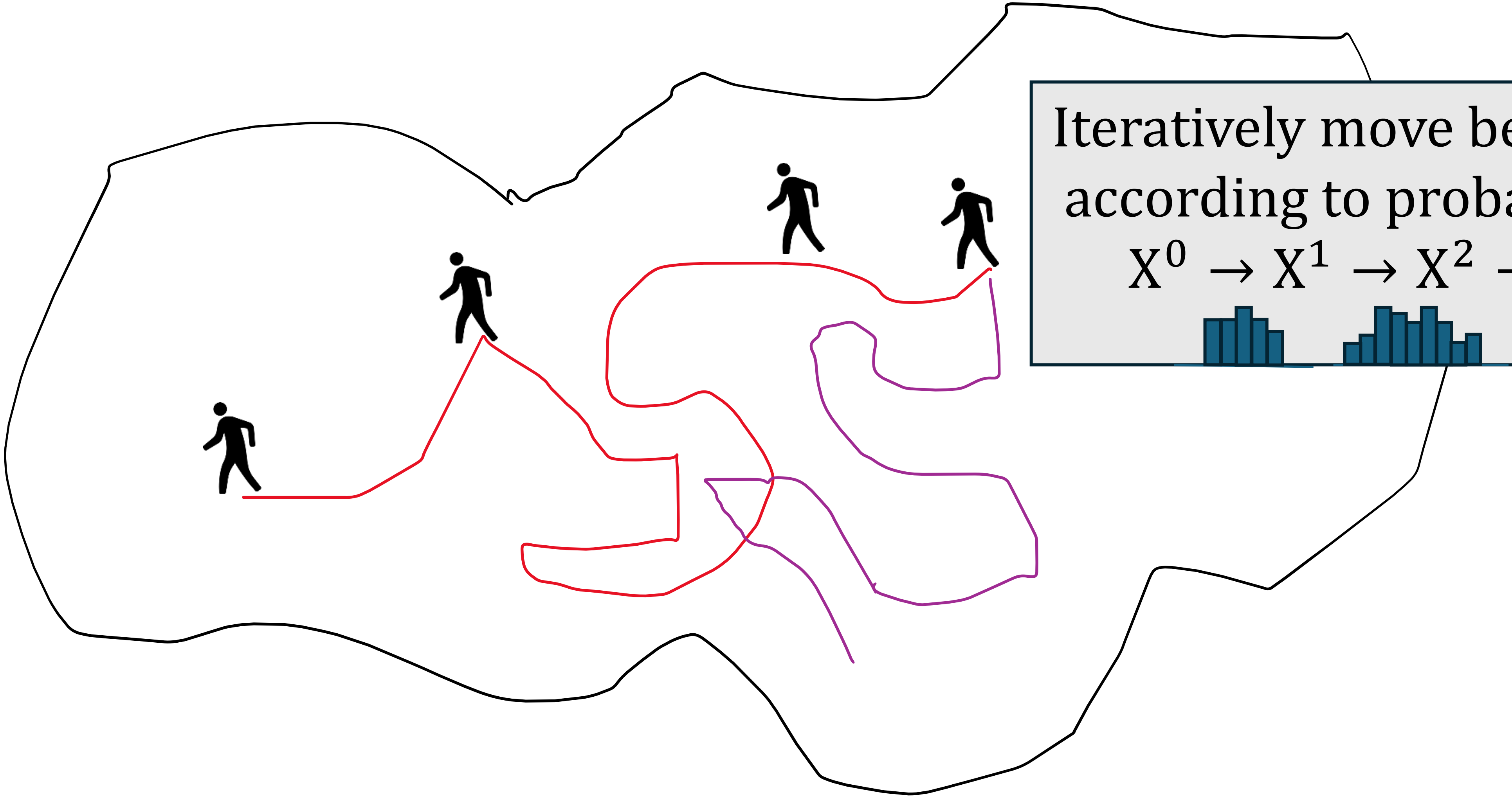
Random walk



Iteratively move between states according to **probabilistic rule:**

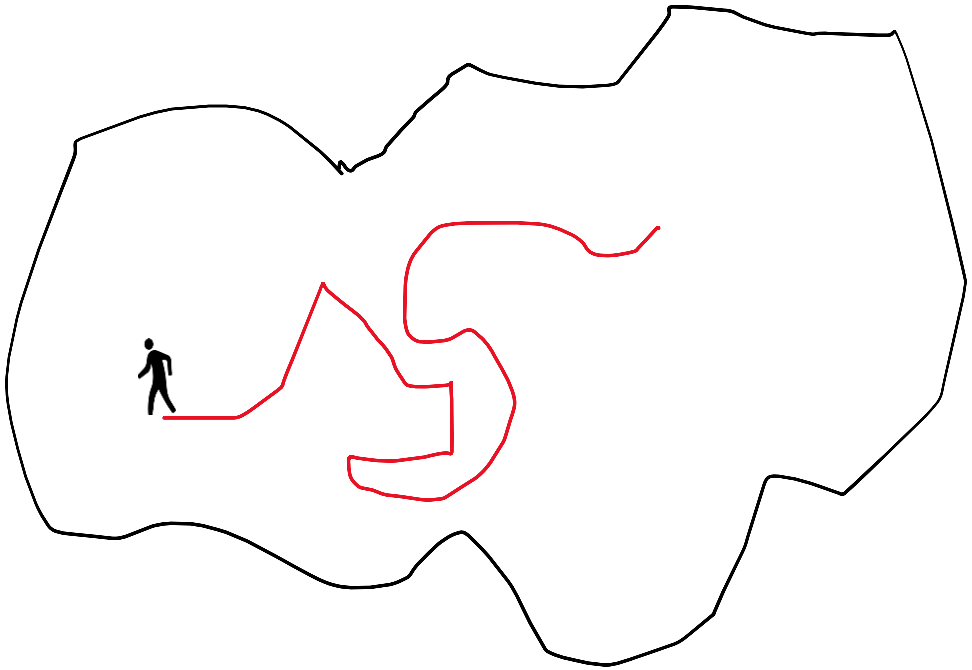
$X^0 \to X^1 \to X^2$ …

Random choices induces sequence of distributions



Iteratively move between states according to probabilistic rule:

$$X^0 \rightarrow X^1 \rightarrow X^2 \rightarrow \ \ldots \rightarrow X^T$$



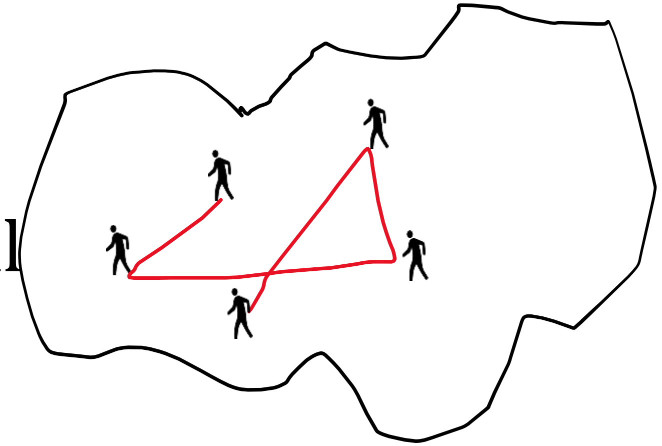
Sampling algorithm:

- Choose transition rule s.t. $X^0 \to X^1 \to X^2 \to \ldots \to X^t \to \cdots \to \pi$ and each step is easy to implement
- Start at arbitrary X^0, do T steps of random walk and output X^T
- Hope: $d_{TV}(X^T, \pi) \leq \epsilon$ and T not too large

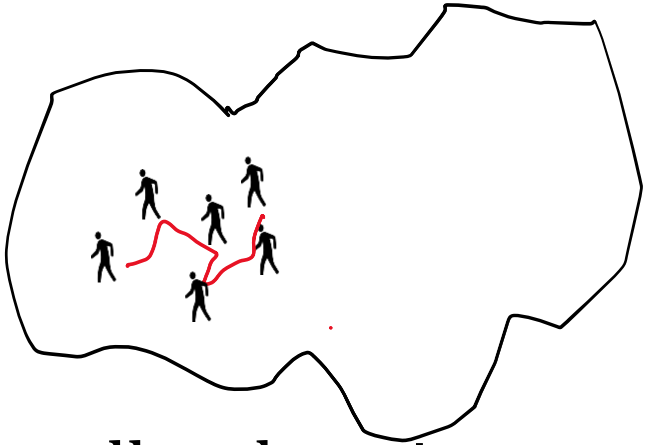
Sampling algorithm:

- Choose transition rule s.t.
  $X^0 \to X^1 \to X^2 \to \ \ldots \to X^t \to \cdots \to \pi$
  and each step is easy (e.g. local)
- Start at arbitrary X^0, do T steps of random walk and output X^T
- Hope: $d_{TV}(X^T, \pi) \leq \epsilon$ and T not too large

Non-local



Local



Local walk ≡ locations at step t and t+1 are close

Issues:

- Don’t directly have access to transition probability in our setting
- For some π, T can be very large

Sampling algorithm:

- Choose transition rule s.t. $X_t \to \pi$ and each step is easy (e.g. local)
- Start at arbitrary X^0, do T steps of random walk and output X^T
- Hope: $d_{TV}(X^T, \pi) \le \epsilon$ and T not too large

Goal: run random walk s.t. $X_t \to \mu$
& each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability

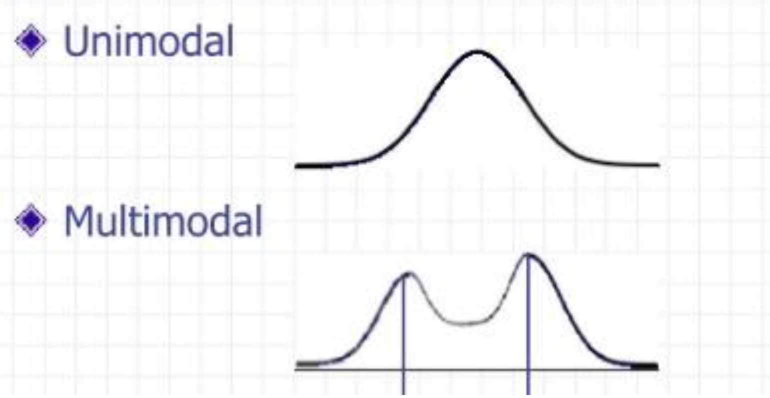
Fix:

- For some RW, can estimate transition probabilities from training data

Goal: run random walk s.t. $X_t \to \mu$
& each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability
- For multimodal π, convergence time T is large



Fix:

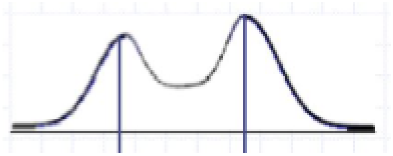
- For some RW, can estimate transition probabilities from training data

- Multimodality due to non-homogeneity
  Example: human height distribution
- Multimodality → slow convergece.

Goal: run random walk s.t. $X_t \to \mu$ & each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability
- For multimodal π, convergence time T is large



- Local walk avoid moving into low-probability regions
- Avoid the valley/bottleneck between peaks
- Cannot cross from one peak to another

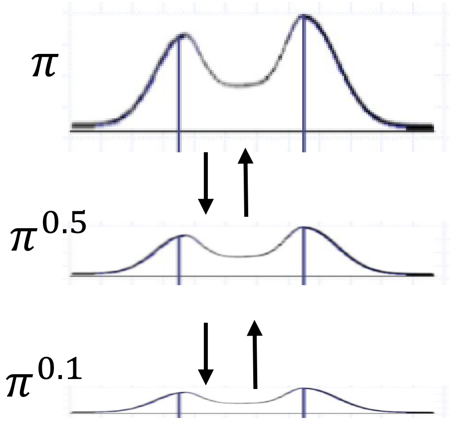
Fix:

- For some RW, can estimate transition probabilities from training data

Goal: run random walk s.t. $X_t \to \mu$ & each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability
- For multimodal π, convergence time T is large



1. Local walk: fails
2. Annealing: fails [GLR'18]

Fix:

- For some RW, can estimate transition probabilities from training data

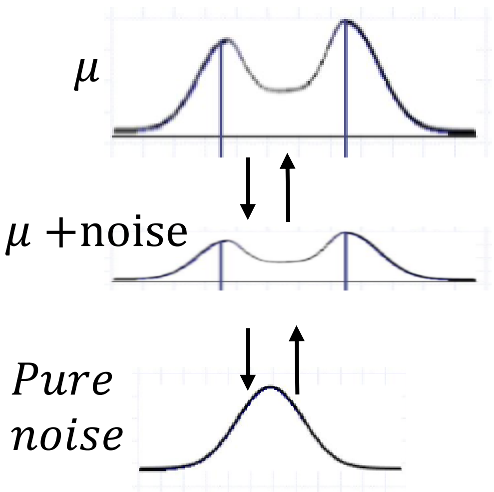
Annealing:

- Removes multimodality by flattening π
- Slow mixing for simple bimodal μ [GLR18]

Goal: run random walk s.t. $X_t \to \mu$
& each step is easy (e.g. local)

Issues:

- Don’t directly have access to transition probability
- For multimodal μ, convergence time T is large



1. Local walk: fails
2. Annealing: fails
3. Denoising diffusion

Fix:

- For some RW, can estimate transition probabilities from training data

Denoising diffusion (DDPM):

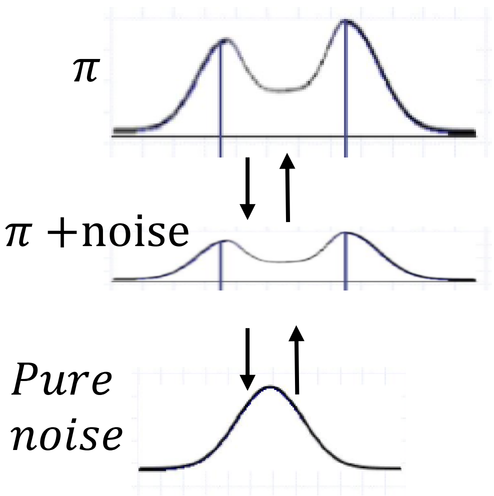
- For continuous distr
- Transition prob. of discrete analog is hard to learn

Goal: run random walk s.t. $X_t \to \mu$ & each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability

- For multimodal π, convergence time T is large



1. Local walk: slow
2. Annealing: slow
3. Denoising diffusion: fast convergence but transition probabilities is hard to learn

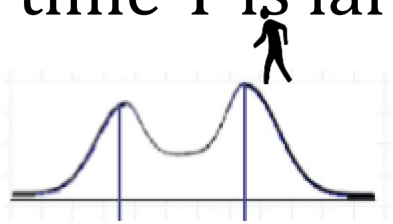
Fix:

- For some RW, can estimate transition probabilities from training data

Goal: run random walk s.t. $X_t \to \mu$ & each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability
- For multimodal π, convergence time T is large



- Local walk cannot move between peaks

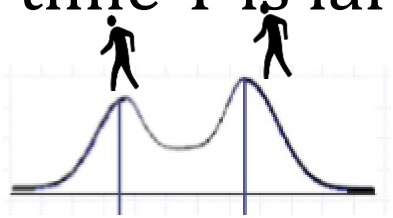
Fix:

- For some RW, can estimate transition probabilities from training data

Goal: run random walk s.t. $X_t \to \mu$
& each step is easy (e.g. local)

Issues:

- Don't directly have access to transition probability
- For multimodal π, convergence time T is large



  - Local walk cannot move between peaks
  - What if we start local walks from all peaks?
  - Average distr. over workers converge to μ very fast

Fix:

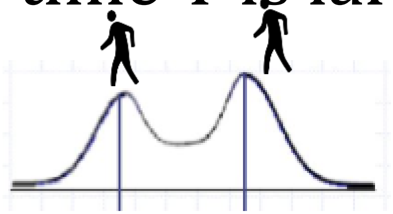
- For some RW, can estimate transition probabilities from training data
- Start local walk from training samples
- Expect to mix fast if #samples is large enough to cover the peaks

Our framework

Issues:

- Don't directly have access to transition probability

- For multimodal π, convergence time T is large



  - Local walk cannot move between peaks
  - What if we start local walks from all peaks?
  - Average distr. over workers converge to μ very fast

Algorithm:

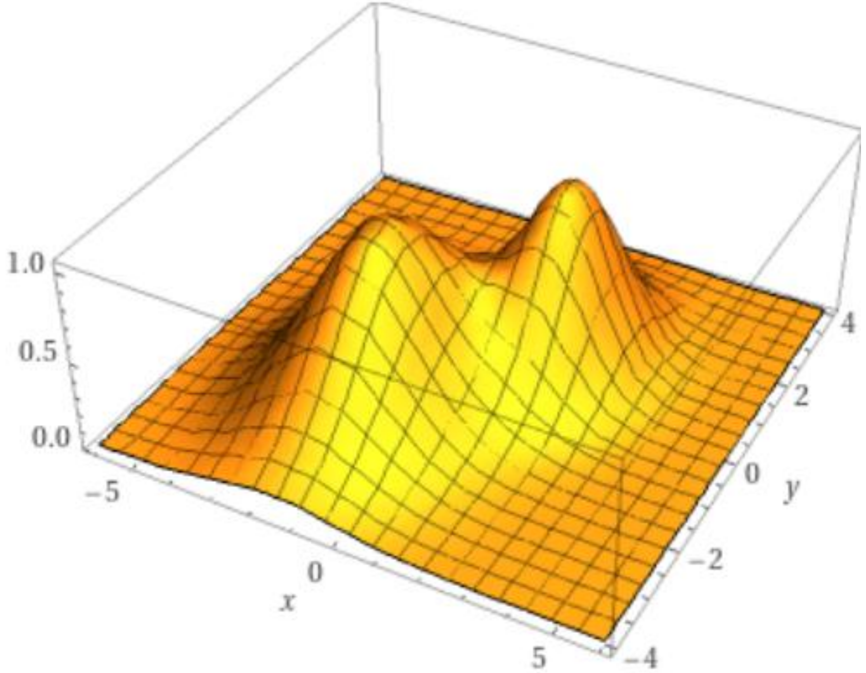
- For local walk, can provably estimate transition probabilities from training data

- Prove that local walk from empirical distribution over training samples converge to π fast if #samples is large enough to cover the peaks

Application

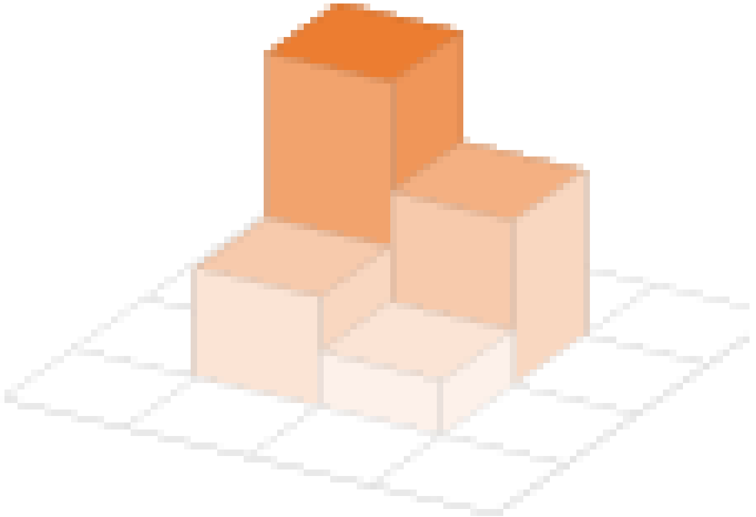
Continuous distribution
$supp(\pi) = \mathbb{R}^d$

Gaussian mixture



Discrete distribution
$supp(\pi) = \{-1, +1\}^d$

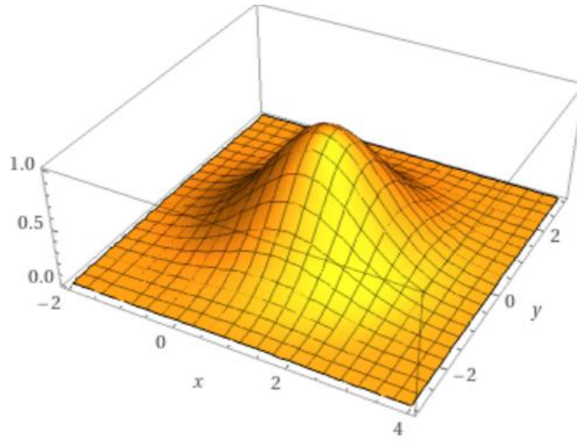
Graphical (Ising) model



Application 1: mixture of Gaussians

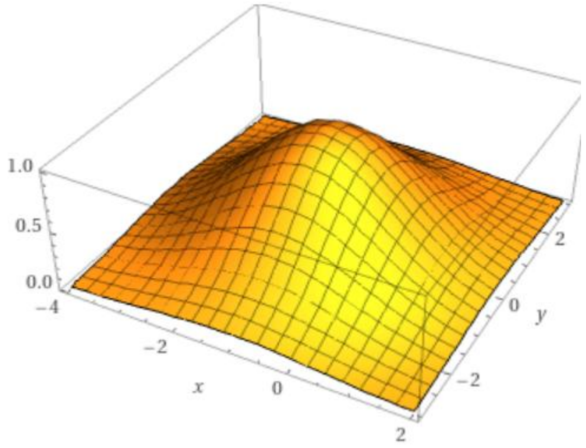
If $\pi = \sum_{i=1}^{k} p_i \pi_i$, $\pi_i \colon \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ is $Gaussian(m_i, \Sigma_i)$

All smooth continuous distribution π ≈ a mixture of Gaussians



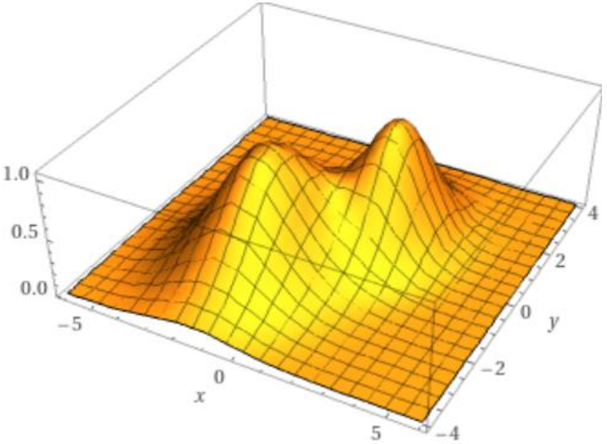
$\pi_1 = Gaussian(m_1, \Sigma_1)$

+



$\pi_2 = Gaussian(m_2, \Sigma_2)$

=



$$\pi = \frac{1}{2}\pi_1 + \frac{1}{2}\pi_2$$

Application 1: mixture of Gaussians

If $\pi = \sum_{i=1}^{k} p_i \pi_i$, $\pi_i: \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ is $Gaussian(m_i, \Sigma_i)$

All smooth continuous distribution $\pi \approx$ a mixture of Gaussians

$k =$ measuring complexity of π

Application 1: mixture of Gaussians

If $\pi = \sum_{i=1}^{k} p_i \pi_i\,, \pi_i : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ is $Gaussian(m_i, \Sigma_i)$

Long-studied testbed for learning & sampling algorithm.

- $k = 1$: [BE'85,Vil'03,VW'19,CELSZ'21]
- $k > 1$:
  - Parameter learning: [Pearson'94,Das'99,SK'01,VW'04,MV'10,HK'13,DS'20,GHK'15]
  - Sampling:
    - ❖ [GLR'18a,b]: Only for isotropic Gaussians, $\Sigma_i = \Sigma \forall i$
    - ❖ [KV23]: For general mixture but has bad runtime dependency on k

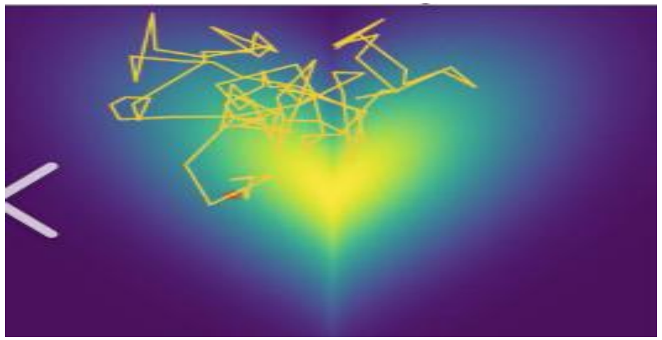
Application 1: mixture of Gaussians

If $\pi = \sum_{i=1}^{k} p_i \pi_i$, $\pi_i : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ is $Gaussian(m_i, \Sigma_i)$, $\alpha I \preccurlyeq \Sigma_i \preccurlyeq \beta I$ then:

$\mu_t \equiv$ *continuous Langevin initialized at* $y_1, \ldots, y_n \sim \pi$ *i.i.d.*
w/ transition probabilities (score function) learned from samples
[Gatmiry-Kelner-Lee'24,Chen-Kontonis-Shah'24]

Continuous Langevin $\equiv$ *Noisy gradient ascent*

$$dX_t = \underbrace{\nabla \log \pi(X_t)}_{\text{score function}} + \underbrace{dB_t}_{\text{Brownian motion}}$$



Application 1: mixture of Gaussians

If $\pi = \sum_{i=1}^{k} p_i \pi_i$, $\pi_i: \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ is $Gaussian(m_i, \Sigma_i)$, $\alpha I \preccurlyeq \Sigma_i \preccurlyeq \beta I$ then:

$\mu_t \equiv$ continuous Langevin initialized at $y_1, \dots, y_n \sim \pi$ i.i.d.
w/ transition probabilities (score function) learned from samples
[Gatmiry-Kelner-Lee'24,Chen-Kontonis-Shah'24]

$d^{poly(\frac{k}{\epsilon_{TV}})}$ samples

Let $n = \Omega\left(\frac{k}{\epsilon_{TV}^2} \log\left(\frac{k}{\rho}\right)\right)$, $T = \frac{\tilde{O}(1)}{\alpha}$

With probability $1 - \rho$, $d_{TV}(\mu_T, \pi) \leq \epsilon_{TV}$

Generalized to mixture of isoperimetric distributions

For $\pi = \sum_{i=1}^{k} p_i \pi_i$ where π_i satisfies log-Sobolev (Poincare resp.) inequality:

- Convergence time is optimal
- Matches convergence time of the case $k = 1$ i.e. π satisfies log-Sobolev (Poincare resp.) inequality
- Robust to perturbation/discretization/score error

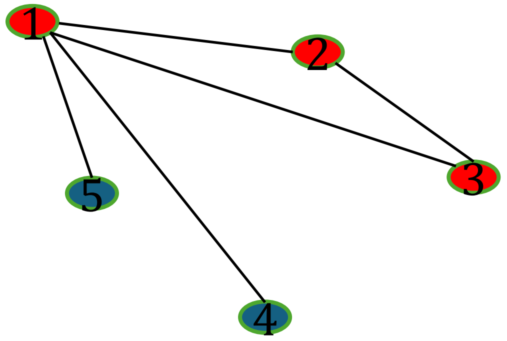
Discussion

For $\pi = \sum_{i=1}^{k} p_i \pi_i$ where π_i satisfies log-Sobolev (Poincare resp.) inequality:

- Convergence time is optimal
- Matches convergence time of the case $k = 1$ i.e. π satisfies log-Sobolev (Poincare resp.) inequality
- Robust to perturbation/discretization/score error
- If π_i's are Gaussians then can estimate transition probabilities of denoising diffusion (DDPM) using [GKL'24,CKS'24], but unclear for general isoperimetric π_i

Application 2: low-complexity (low-rank) Ising

Ising model $\pi: \{\pm 1\}^n \to \mathbb{R}_{\geq 0}, \pi(x) \propto \exp(\frac{1}{2}\langle x, Jx\rangle + \langle h, x\rangle)$:



$X_1, \cdots, X_n$ are random variables

- J_{ij} encodes correlation of X_i, X_j
- h_i encodes bias of X_i

Motivation:

- Simplest discrete distribution with non-trivial correlations
- Hopfield network [Lit74,Hop82,PF77]
- Stochastic block model [Sin11,DAM17,AMM+18]
- Bayesian inference in linear regression [DAM17, LM19, MV21,MW24]

Motivation: Bayesian inference in linear regression

[DAM17, LM19, MV21,MW24]

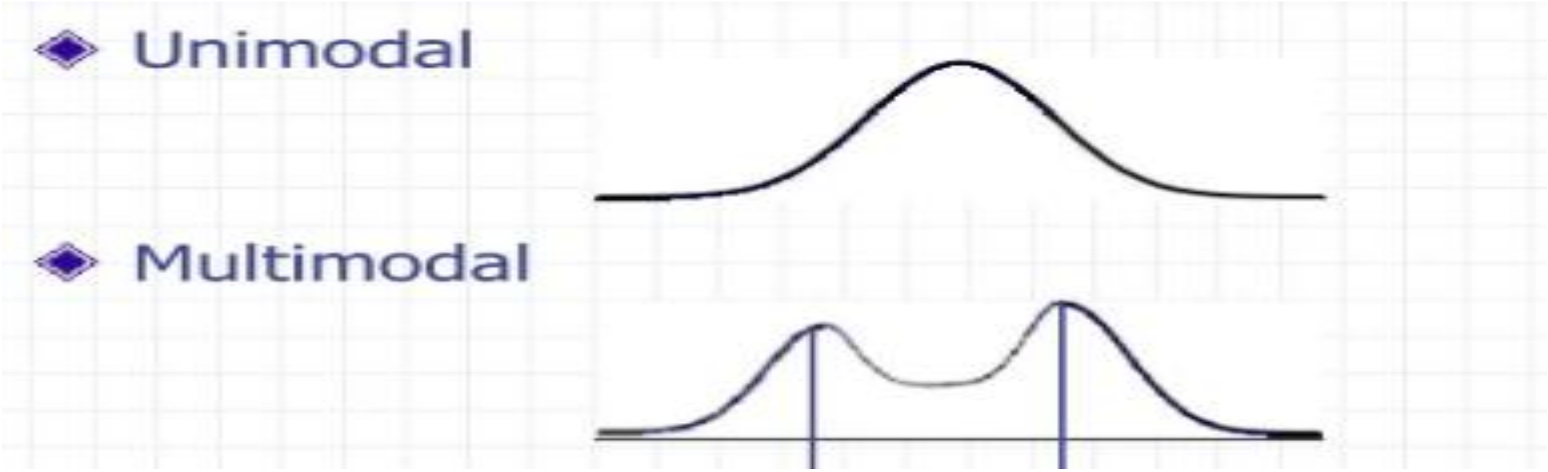
Given observation $y_0 = X\Theta + Gaussian(0, \sigma^2 I)$,
the Bayesian estimator for Θ with prior Uniform($\{\pm 1\}^{\mathrm{n}}$) is

$$\pi(\theta) \propto \exp\left(-\frac{||y_0 - X\Theta||^2}{2\sigma^2}\right) = \text{Ising with } J = X^T X/\sigma^2 \text{ and } h = y_0^T X/2\sigma^2$$

Note:

- J is PSD
- $\mathrm{Rank}(J) = \dim(y_0) \ll \mathrm{n}$

Multimodality of Ising model



$\lambda < 1$

$\lambda > 1$

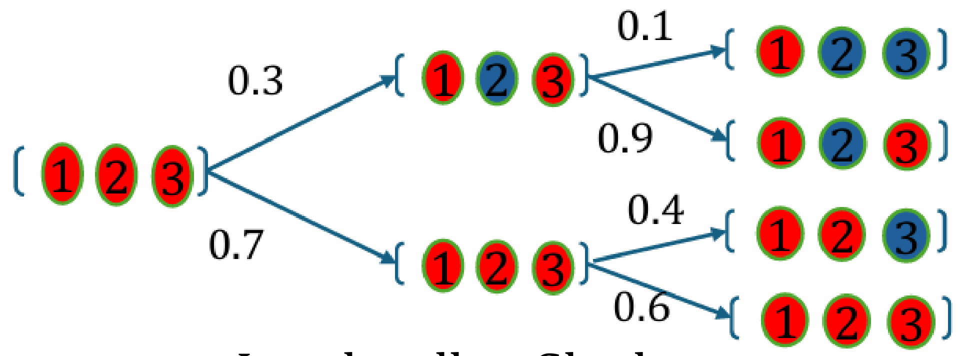
Projection of Ising model with $J = \lambda u u^T, ||u|| = 1$ to 1-dimension

Application 2: low-complexity (low-rank) Ising

Ising model $\pi: \{\pm 1\}^n \to \mathbb{R}_{\geq 0}, \pi(x) \propto \exp(\frac{1}{2}\langle x, Jx\rangle + \langle h, x\rangle)$:

≈Low-rank { Eigenvalues of J: $\lambda_1 \geq \cdots \geq \lambda_r > 1 - \frac{1}{c} \geq \lambda_{r+1} \geq \cdots \geq \lambda_n$
s.t. sum (negative eigenvalues) $\leq O(1)$ }

$\mu_t \equiv$ *Glauber initialized at* $y_1, \ldots, y_n \sim \pi$ *i.i.d.*
with transition probabilities learned from $y_1, \ldots, y_n$ *via pseudo-likelihood [Bes75]*



Local walk—Glauber:
each step resamples 1 location

Application 2: low-complexity (low-rank) Ising

Ising model $\pi: \{\pm 1\}^n \to \mathbb{R}_{\geq 0}, \pi(x) \propto \exp(\frac{1}{2}\langle x, Jx\rangle + \langle h, x\rangle)$:

≈Low-rank: Eigenvalues of J: $\lambda_1 \geq \cdots \geq \lambda_r > 1 - \frac{1}{c} \geq \lambda_{r+1} \geq \cdots \geq \lambda_n$
s.t. sum (negative eigenvalues) $\leq O(1)$

$\mu_t \equiv$ *Glauber initialized at* $y_1, \ldots, y_n \sim \pi$ *i.i.d.*
with transition probabilities learned from $y_1, \ldots, y_n$ *via pseudo-likelihood*

Let $n = \Omega\left((nr\lambda_1)^{O(r)} \log(\frac{1}{\rho})/\epsilon_{TV}^4\right), T = \tilde{O}(n\lambda_1)$

With probability $1 - \rho$, $d_{TV}(\mu_T, \pi) \leq \epsilon_{TV}$

Discussion

Ising model $\pi: \{\pm 1\}^n \to \mathbb{R}_{\geq 0}, \pi(x) \propto \exp(\frac{1}{2}\langle x, Jx\rangle + \langle h, x\rangle)$:

≈Low-rank {
Eigenvalues of J: $\lambda_1 \geq \cdots \geq \lambda_r > 1 - \frac{1}{c} \geq \lambda_{r+1} \geq \cdots \geq \lambda_n$
s.t. sum (negative eigenvalues) $\leq O(1)$

- If $r = O(1)$, new efficient (distribution) learner
- Separation between parameter learning & distribution learning

parameter learning: $\Omega(\exp(n))$ samples

distribution learning: $poly(n)$ samples

Proof

Challenge

- Most analysis techniques only handle convergence time from worst-case start

Challenge

- Most analysis techniques only handle convergence time from worst-case start
- Exceptions:
  - Glauber on symmetric Ising & related models [GS22; BGZ24; BMP21; Cuf+12; LLP10; DLP09a; DLP09b; GGS24]: exploit special properties in stat. physics setting (symmetricity, monotonicity)

Challenge

- Most analysis techniques only handle convergence time from worst-case start
- Exceptions:
  - Glauber on symmetric Ising & related model
  - Langevin on Gaussian mixtures [KV23]: bad dependency on $k =$ #components.
    Can only bound convergence time $T \leq 2^{2^k}$ since:
    - ❖ Proof looks at how component overlaps,
    - ❖ Becomes very complicated as the overlaps structure has exponential dependency on k

This work

- Tight bounds and exponentially improve on [KV23]
- Unifying proof for continuous and discrete distributions
- Reduce to higher eigenvalue gap

- Most analysis techniques only handle convergence time from worst-case start
- Previous Exceptions:
  - Glauber on symmetric Ising/Potts model: exploit special properties
  - Langevin on Gaussian mixtures [KV23]: Bad dependency on #components due to overlapping analysis

Mixing time and eigenvalues of Markov transition matrix

Transition probability matrix P: $P(x, y) = \mathbb{P}[X_{t+1} = y | X_t = x]$
Eigenvalues of P: $1 = \lambda_1 \geq \lambda_2 \geq \cdots$

Thm (classical): mixes in $\approx \frac{1}{1-\lambda_2}$ steps from worst case start

Fast mixing from empirical sample under higher−order eigenvalue gap

Transition probability matrix P: $P(x, y) = \mathbb{P}[X_{t+1} = y | X_t = x]$
Eigenvalues of P: $1 = \lambda_1 \geq \lambda_2 \geq \cdots$

Thm (classical): mixes in $\approx \frac{1}{1-\lambda_2}$ steps from worst case start

Thm (this work): mixes in $\approx \frac{1}{1-\lambda_k}$ steps when starts at a randomly chosen y_i among $n \approx k$ samples $y_1, \ldots, y_n \sim \pi$

Higher order eigenvalue gap of mixtures

Transition probability matrix P: $P(x, y) = \mathbb{P}[X_{t+1} = y | X_t = x]$
Eigenvalues of P: $1 = \lambda_1 \geq \lambda_2 \geq \cdots$

Thm (this work): mixes in $\approx \frac{1}{1-\lambda_k}$ steps when starts at a randomly chosen y_i among $n \approx k$ samples $y_1, \ldots, y_n \sim \pi$
Thm (this work): $\pi = \sum_{i=1}^{k} p_i \pi_i$ and 2nd-eig of Glauber/Langevin for $\pi_i \leq 1 - \sigma$ then k-th eig of Glauber/Langevin for $\pi \leq 1 - \sigma$

Application:

- Mixture of Gaussians/isoperimetric continuous distribution
- Low-rank Ising ≈ mixture of high-temperature Isings with second eigenvalue gap [KLR22,AKV24]

Fast mixing from empirical sample under higher–order eigenvalue gap

Transition probability matrix P: $P(x,y) = \mathbb{P}[X_{t+1} = y | X_t = x]$
Eigenvalues of P: $1 = \lambda_1 \geq \lambda_2 \geq \cdots$ with eigenvectors $1 \equiv f_1, f_2, \ldots$

Thm (this work): mixes in $\approx \frac{1}{1-\lambda_k}$ steps when starts at a randomly chosen y_i among $n \approx k$ samples $y_1, \ldots, y_n \sim \pi$

Lem:

$\mu_0 \equiv$ initialization. If $\sum_{i=1}^{k} ||\langle \mu_0, f_i \rangle \quad ||^2 \leq \epsilon^2$, $\mathrm{t} = \frac{\log\left(\frac{1}{\epsilon}\right)}{1-\lambda_k}$

$$d_{TV}(\mu_t, \pi) \leq \epsilon$$

Fast mixing from empirical sample under higher−order eigenvalue gap

Transition probability matrix P: $P(x, y) = \mathbb{P}[X_{t+1} = y | X_t = x]$
Eigenvalues of P: $1 = \lambda_1 \geq \lambda_2 \geq \cdots$ with eigenvectors $1 \equiv f_1, f_2, \ldots$

Thm (this work): mixes in $\approx \frac{1}{1-\lambda_k}$ steps when starts at a randomly chosen y_i among $n \approx k$ samples $y_1, \ldots, y_n \sim \pi$

Lem: $\mu_0 \equiv$ initialization. If $\sum_{i=1}^{k-1} ||\langle \mu_0, f_i \rangle \quad ||^2 \leq \epsilon^2$, t $= \frac{\log\left(\frac{1}{\epsilon}\right)}{1-\lambda_k}$, $d_{TV}(\mu_t, \pi) \leq \epsilon$

Lem: $\mu_0 \equiv \frac{1}{n} \sum \delta_{y_i}$. If $n \geq \frac{k}{\epsilon^2} \log\left(\frac{k}{\rho}\right)$ then w. prob $1 - \rho$

$$\sum_{i=1}^{k} ||\langle \mu_0, f_i \rangle \quad |\Big|^2 \leq \epsilon^2$$

Fast mixing from empirical sample under higher−order eigenvalue gap

Transition probability matrix P: $P(x,y) = \mathbb{P}[X_{t+1} = y | X_t = x]$
Eigenvalues of P: $1 = \lambda_1 \geq \lambda_2 \geq \cdots$ with eigenvectors $1 \equiv f_1, f_2, \ldots$

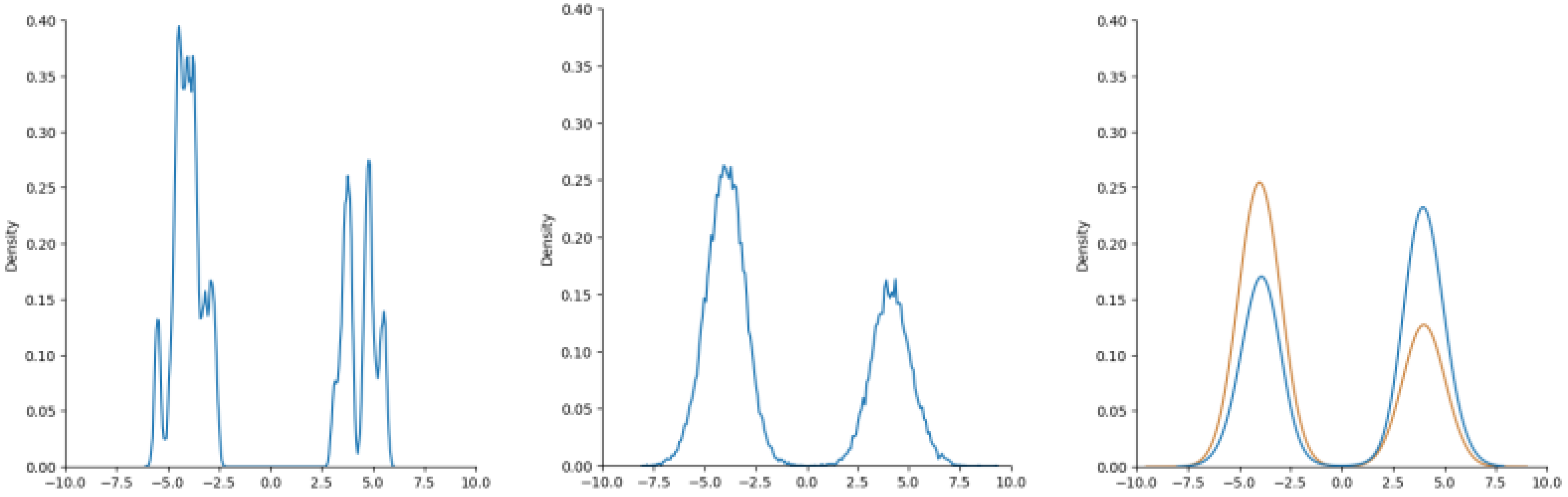
Lem.: $\mu_0 \equiv \frac{1}{n}\sum \delta_{y_i}$. If $n \geq \frac{k}{\epsilon^2}\log\left(\frac{k}{\rho}\right)$ then w. prob $1-\rho$, $\sum_{i=1}^{k-1} ||\langle \mu_0, f_i \rangle \quad ||^2 \leq \epsilon^2$

$$\mathbb{E}_{y\sim\pi}\left[\left\langle \delta_y, f_i \right\rangle\right] = \langle \pi, f_i \rangle = 0$$

$$\mathbb{E}_{y\sim\pi}\left[\left\langle \delta_y, f_i \right\rangle^2\right] = \langle f_i, f_i \rangle = 1$$

We could use Chebyshev, but only get $n \geq \frac{k}{\epsilon^2 \rho}$

New trick: restrict to y with bounded $|\left\langle \delta_y, f_i \right\rangle|$ and use Bernstein+ triangle ineq. to deal with remaining y



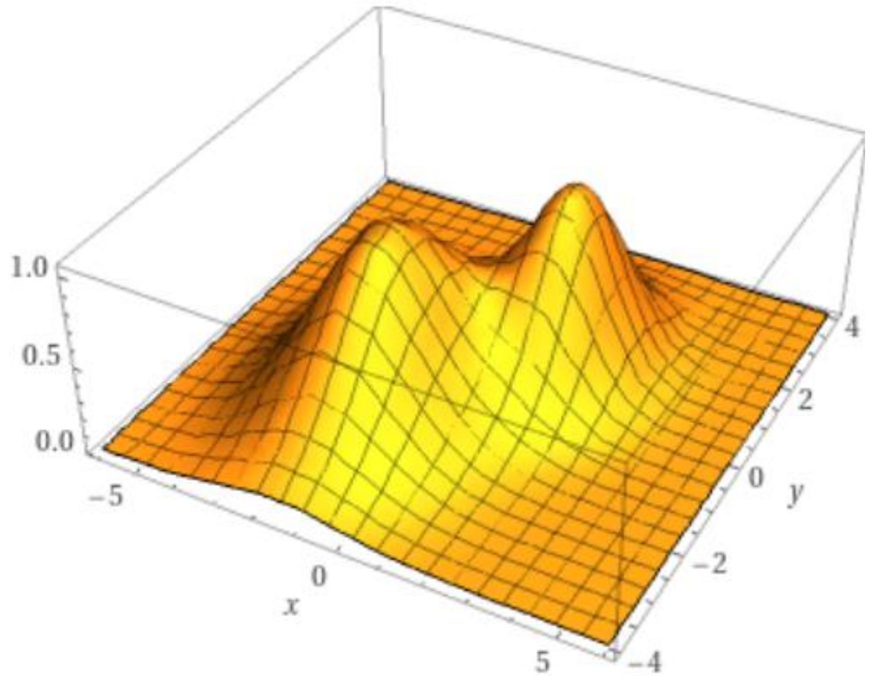
(a) $T = 0$ (b) $T = 200$ (c) $T = \infty$ & truth (orange)

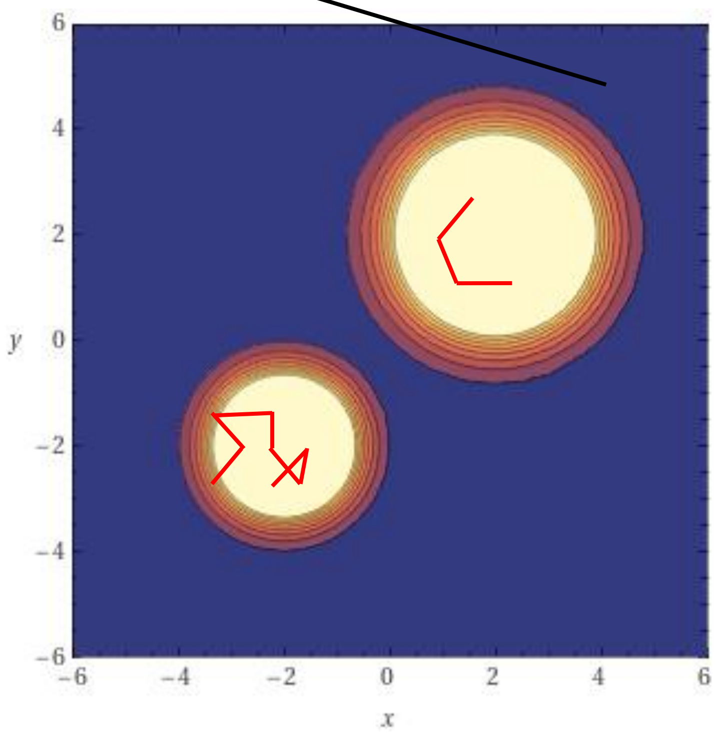
Figure 1: Visualization of the distribution of the Langevin dynamics after T iterations when initialized at the empirical distribution and run with an approximate score function estimated from data. Orange density (rightmost figure) is the ground truth mixture of two Gaussians; the empirical distribution (leftmost figure, $T = 0$) consists of 40 iid samples from the ground truth. Langevin dynamics with step size 0.01 is run with an estimated score function, which was fit using vanilla score matching with a one hidden-layer neural network trained on fresh samples; densities (blue) are visualized using a Gaussian Kernel Density Estimate (KDE). Matching our theory, we see that the ground truth is accurately estimated at time $T = 200$ even though it is not at $T = 0$ or ∞.

Langevin with data-based initialization:

$X_0 = x \sim \text{Uniform}(\{\text{training samples}\})$;

$X_{(n+1)h} - X_{nh} = \nabla \log \mu(X_{nh})h + \mathcal{N}(0,2h)$





Trajectories of Langevin initialized at training samples x_1, x_2