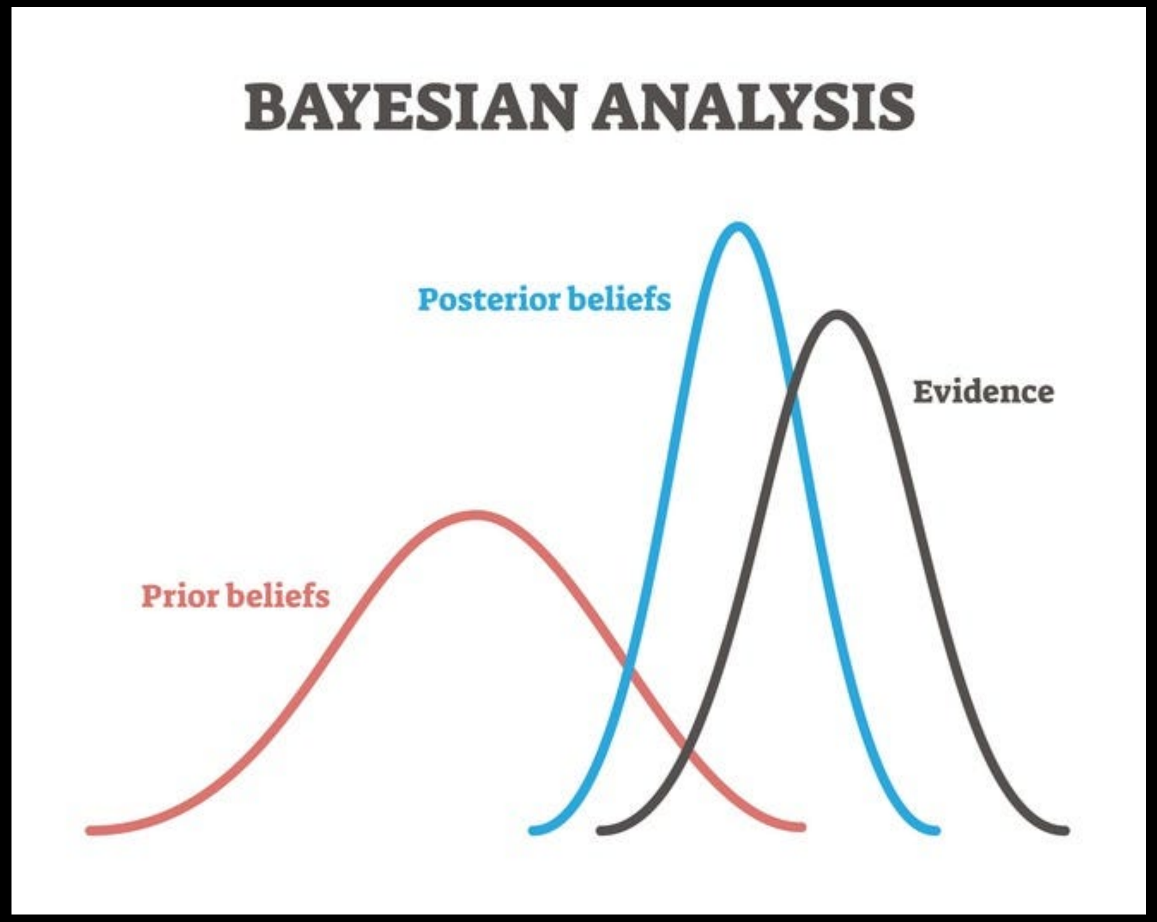
Reverse Diffusion Sequential Monte Carlo Samplers

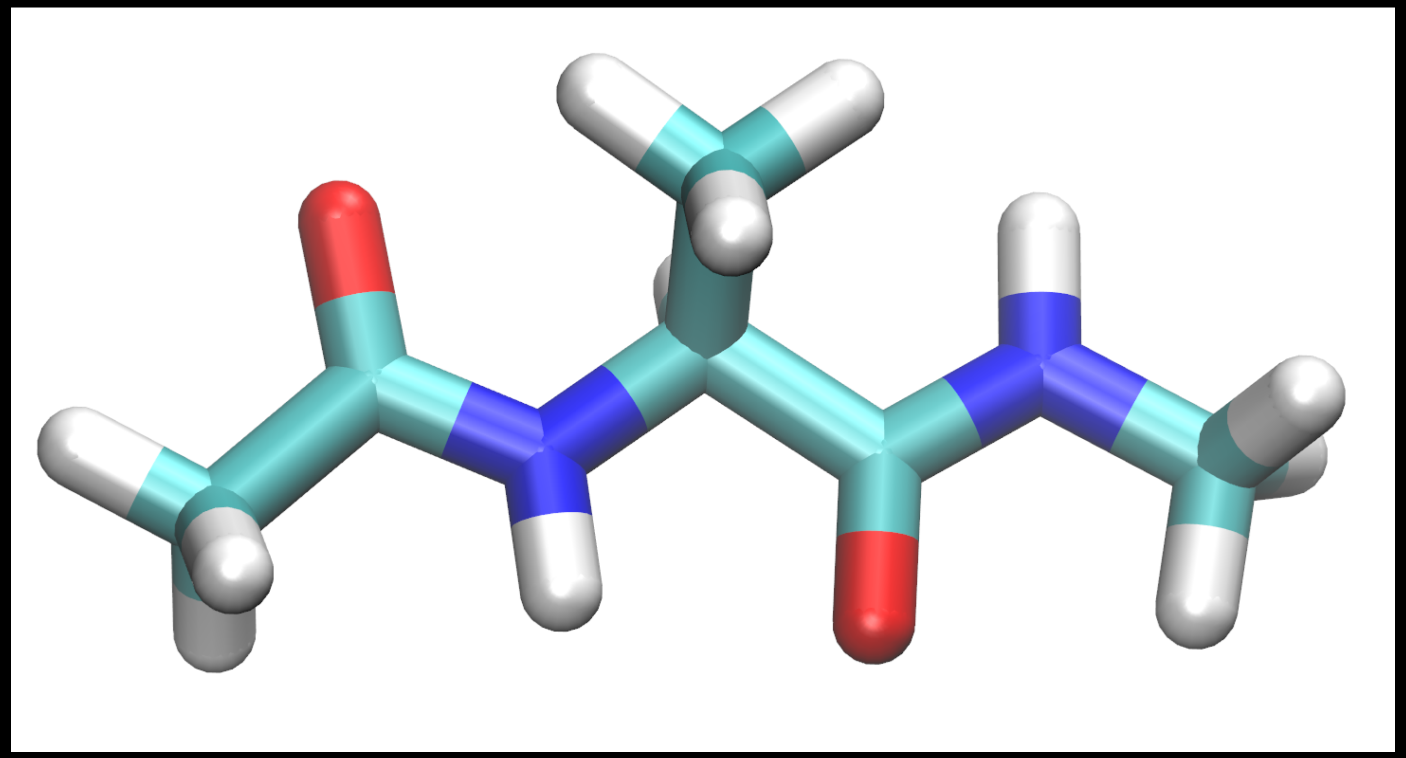


Luhuan Wu Yi Han Christian Naesseth John Cunningham

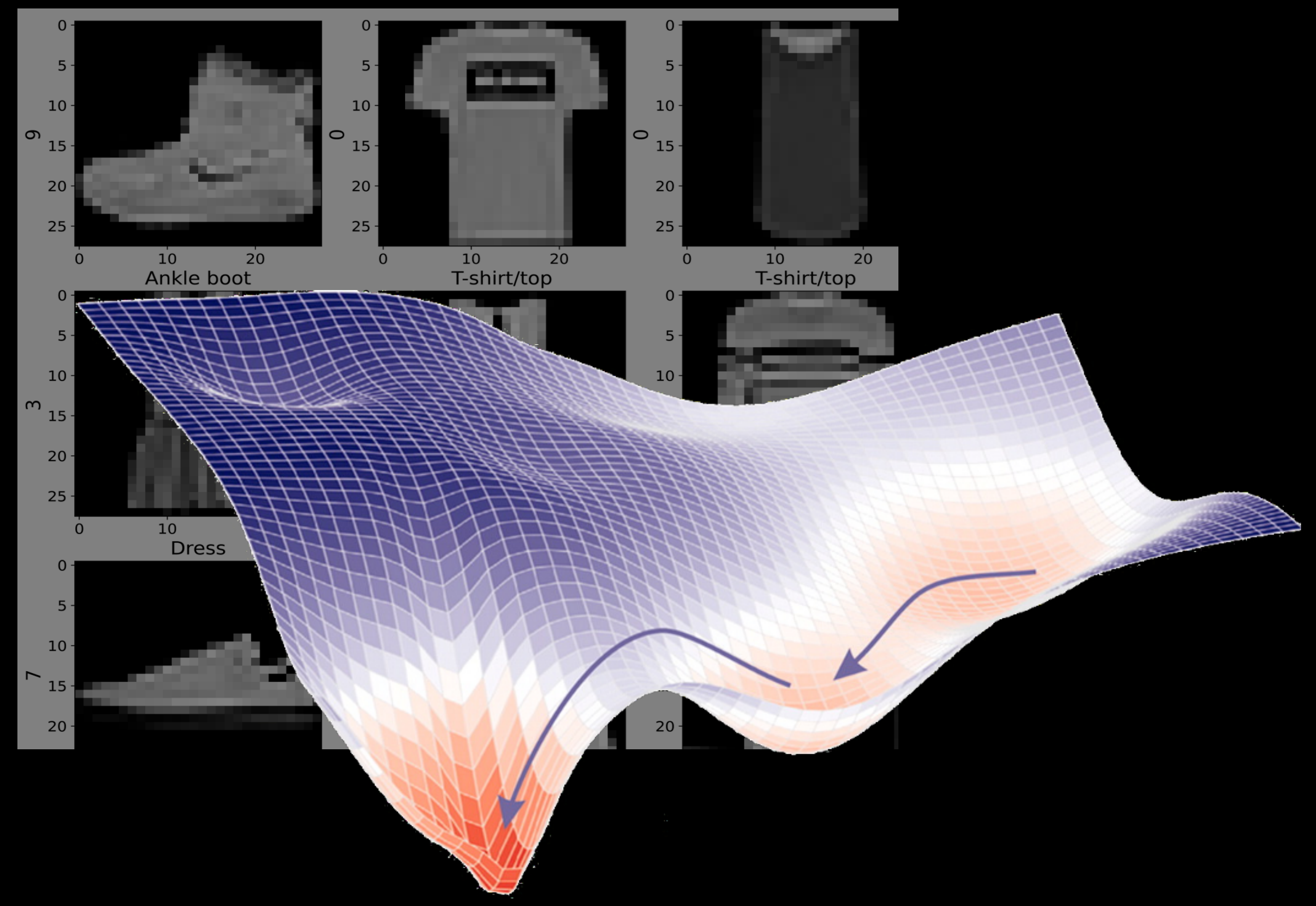
Sampling from unnormalized target $\pi(x) \propto \tilde{\pi}(x)$



Bayesian inference

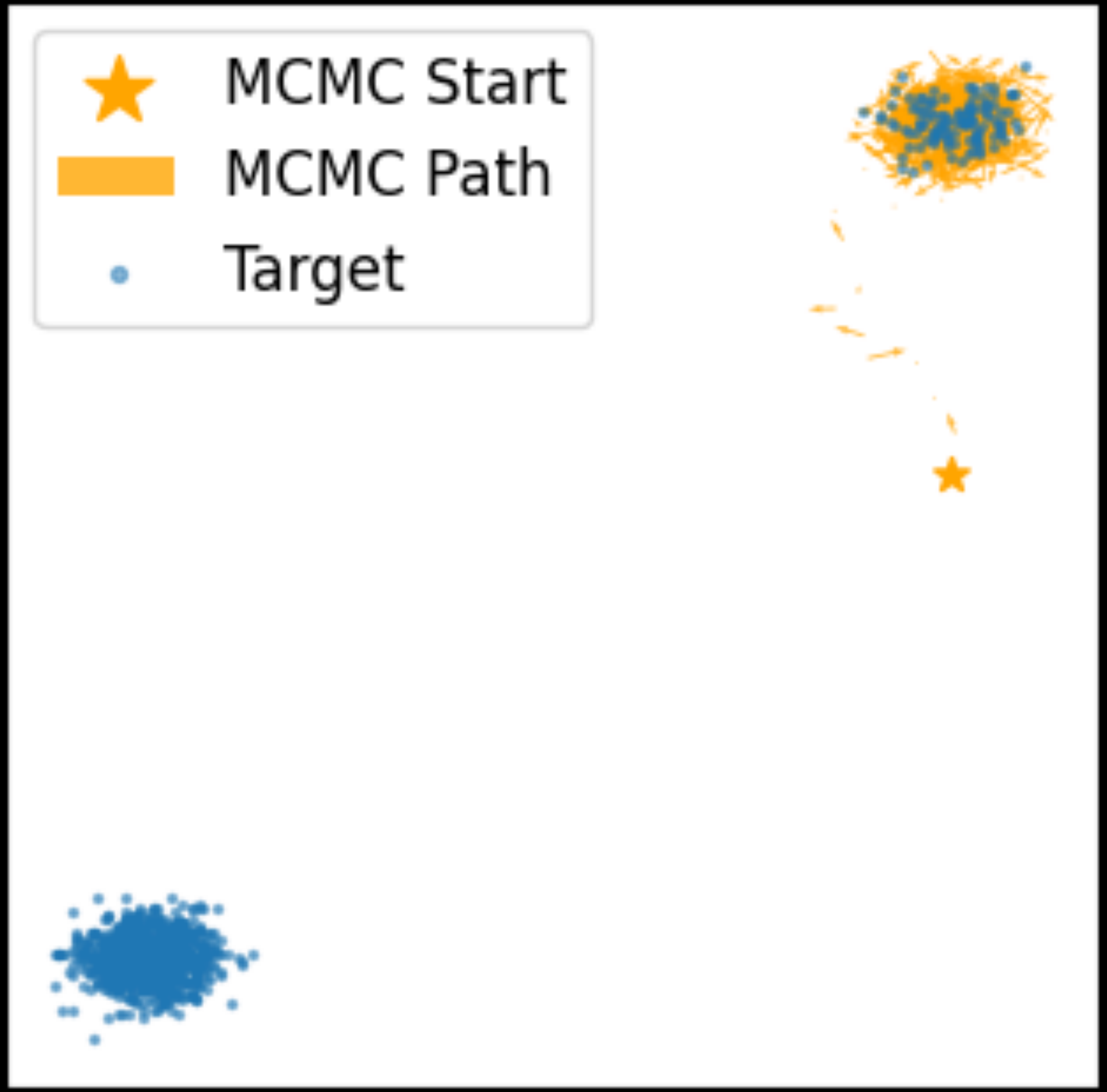


Computational chemistry



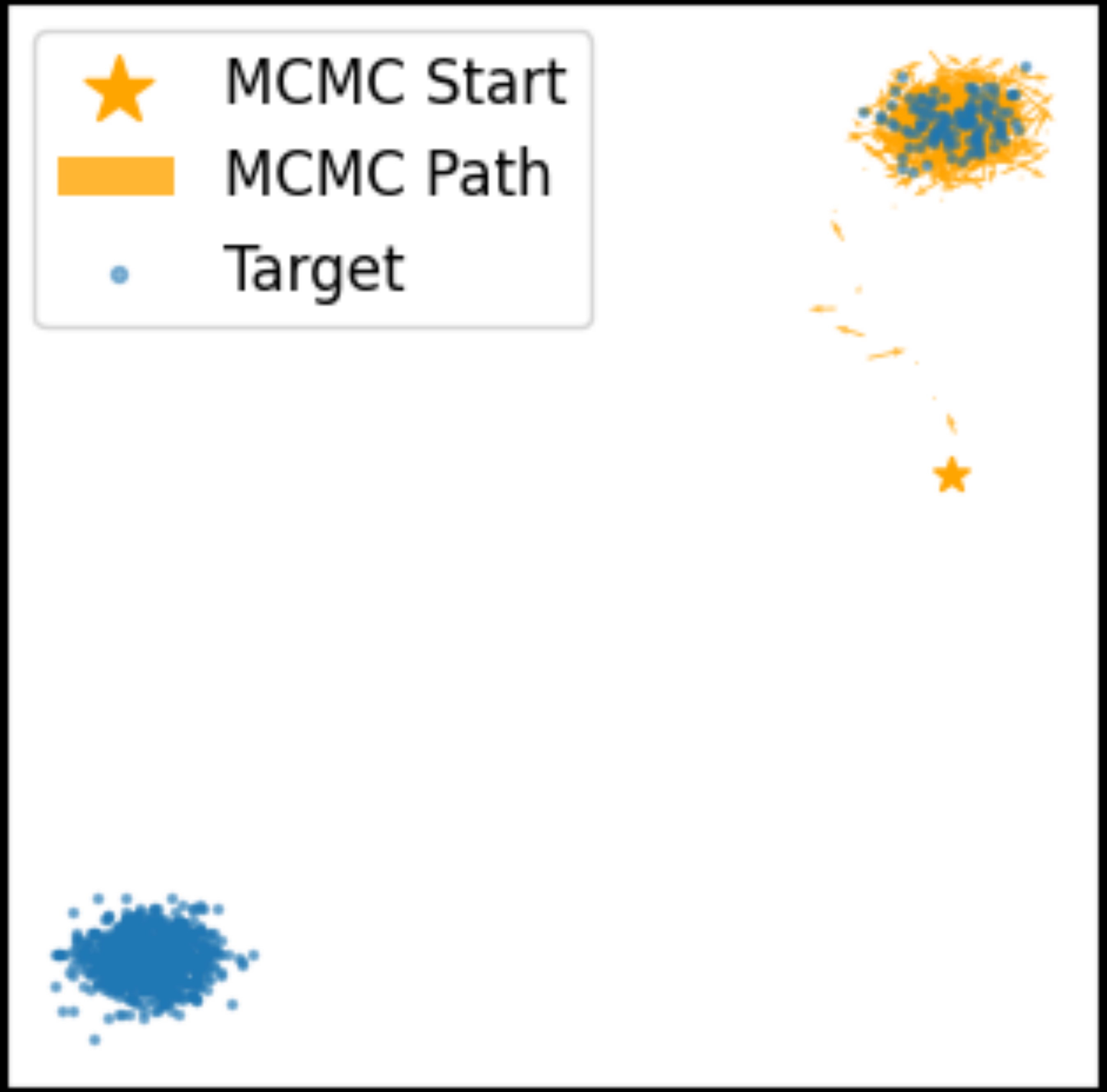
Energy based modeling

Motivation



Challenge: sampling from multi-modal distribution

Motivation



Challenge: sampling from multi-modal distribution

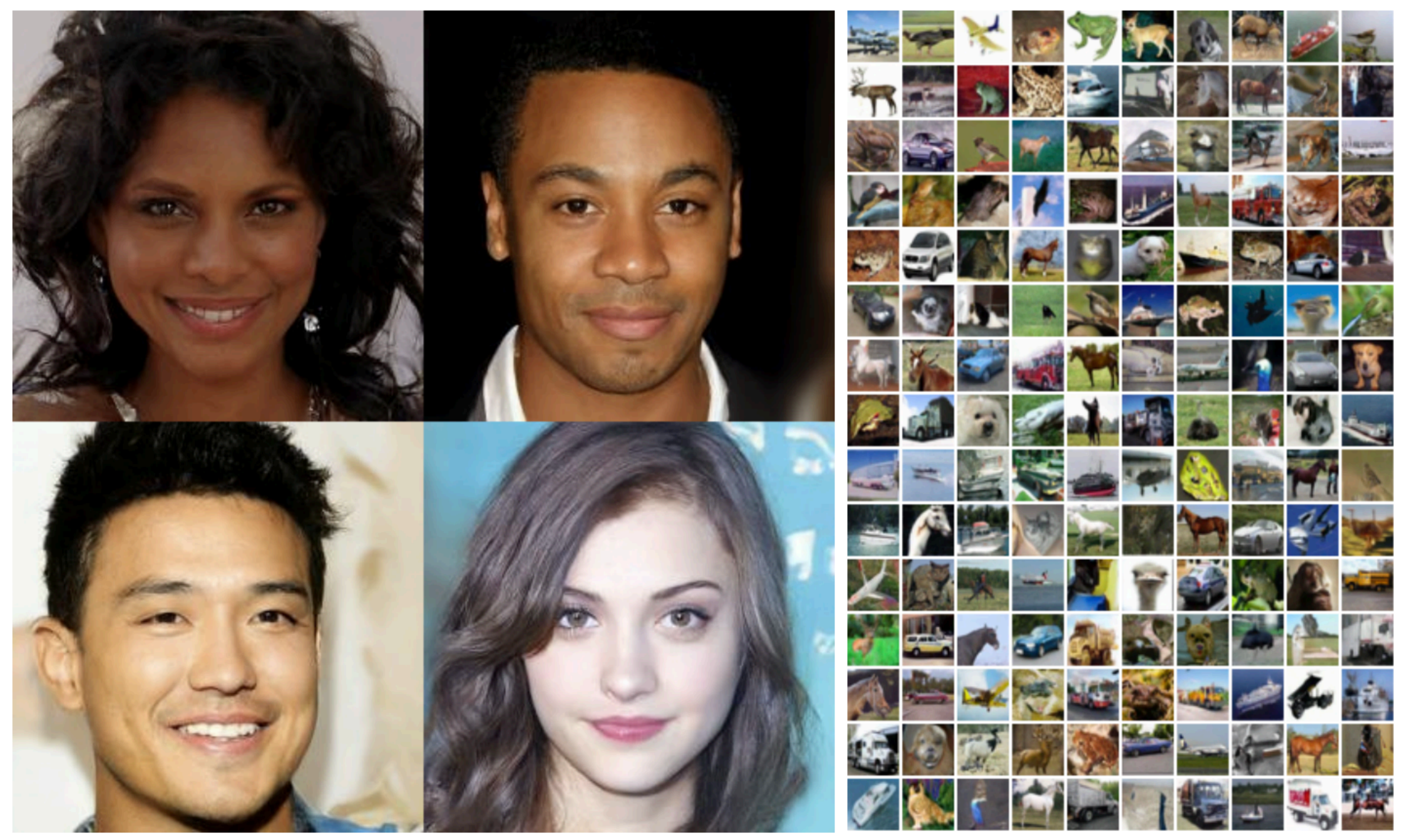


Figure 1: Generated samples on CelebA-HQ 256 × 256 (left) and unconditional CIFAR10 (right)

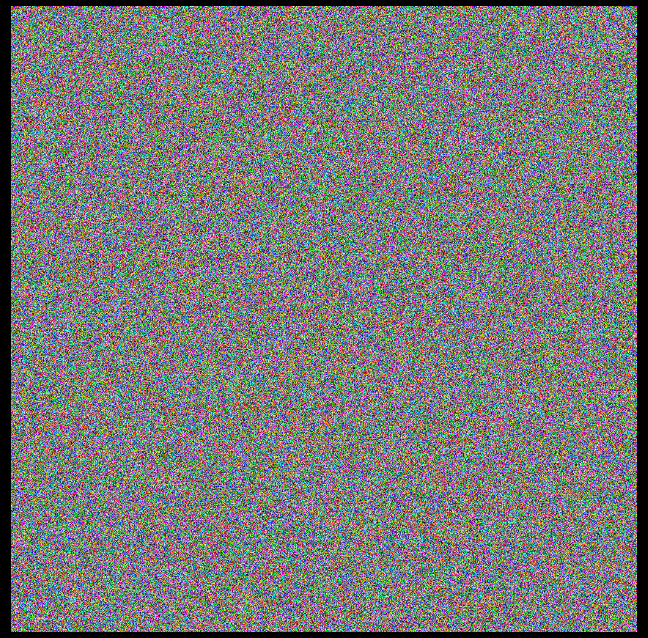
[Ho et al, 2020]

Diffusion models learn complex, multi-modal distributions

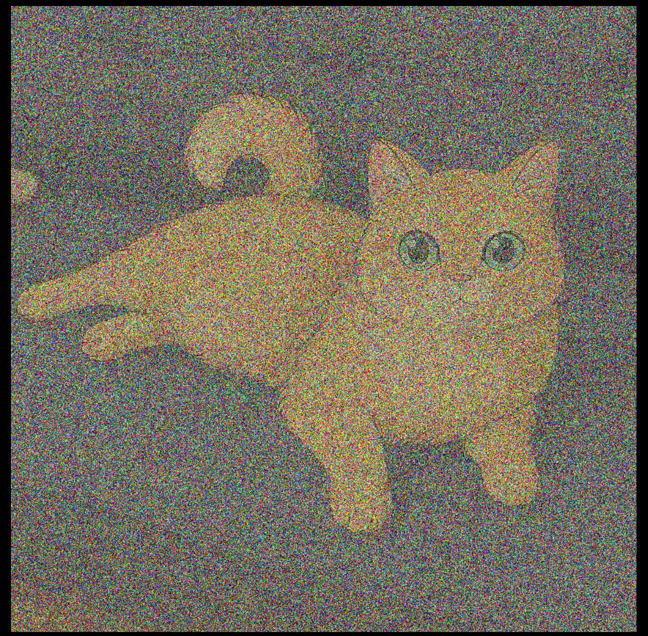
Diffusion models

forward process $dx_t = f_t x_t dt + g_t dB_t$

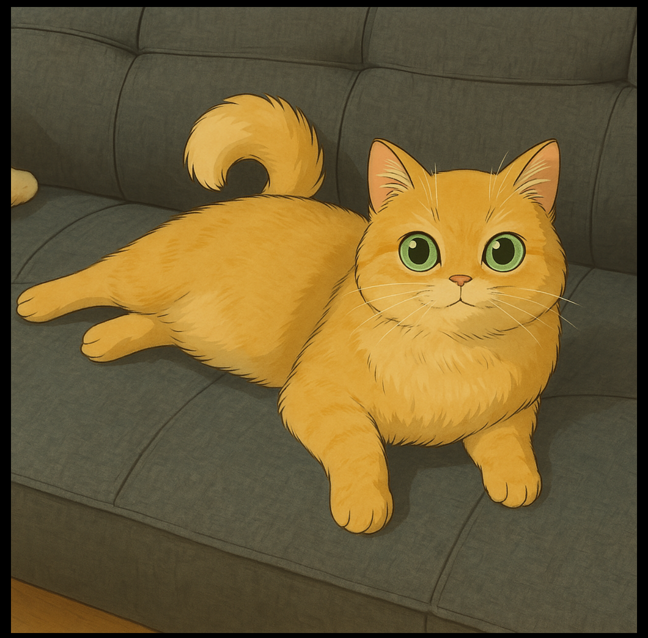
$x_1 \sim \pi(x_1) = \mathcal{N}(0,1)$



$x_t \sim \mathcal{N}(x_t \mid \alpha_t x_0, \sigma_t^2)$



$x_0 \sim \pi(x_0)$



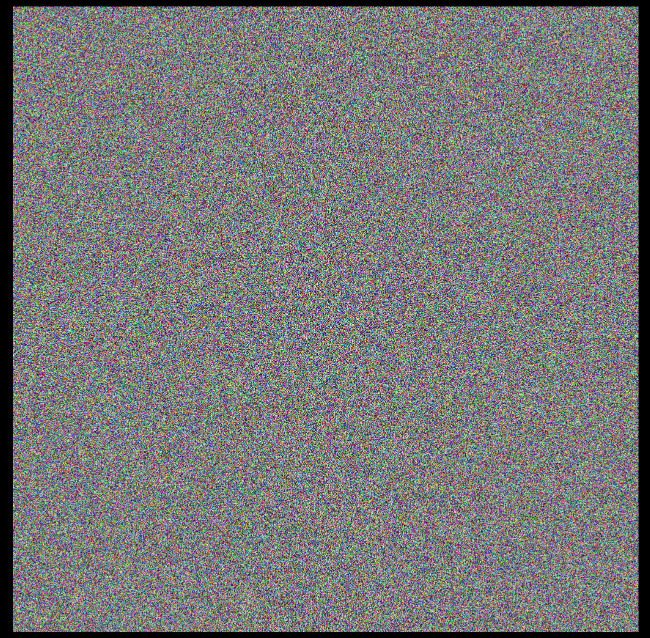
reverse process $dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$

scored function learned from data, which is unavailable in sampling problems😒

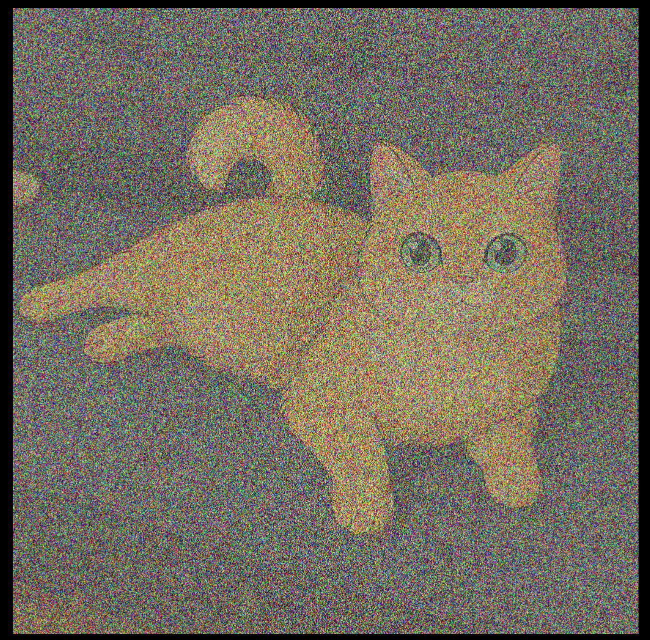
Diffusion models

forward process $dx_t = f_t x_t dt + g_t dB_t$

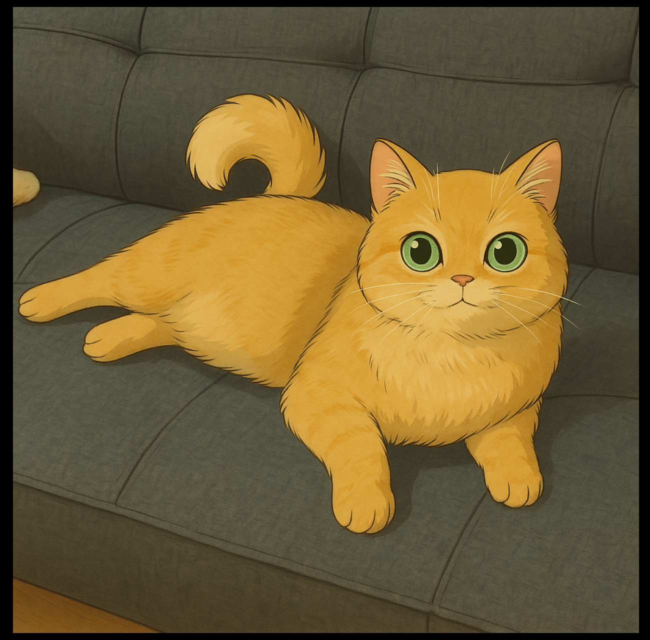
$x_1 \sim \pi(x_1) = \mathcal{N}(0,1)$



$x_t \sim \mathcal{N}(x_t \mid \alpha_t x_0, \sigma_t^2)$



$x_0 \sim \pi(x_0)$



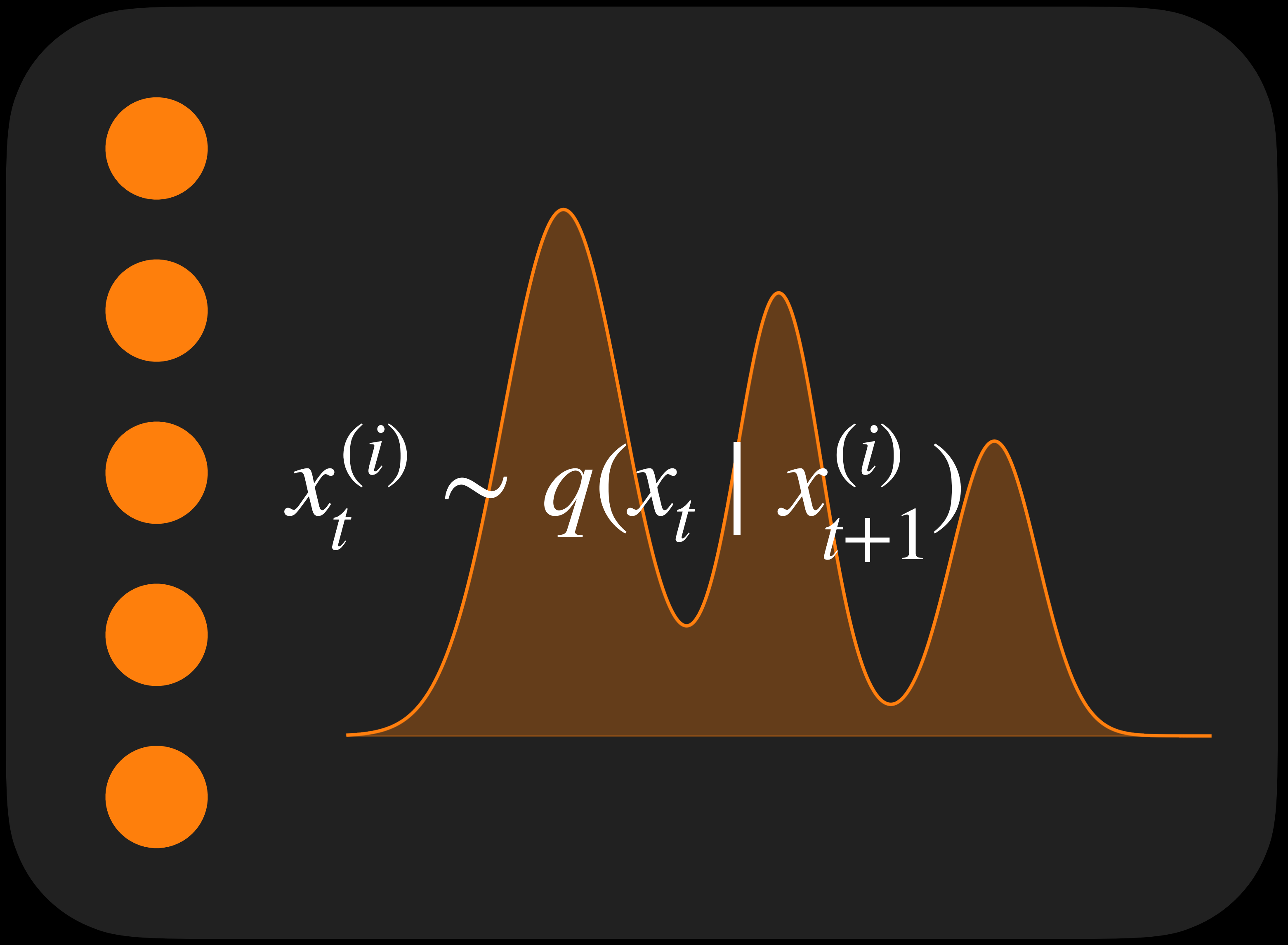
reverse process $dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$

scored function learned from data, which is unavailable in sampling problems😒

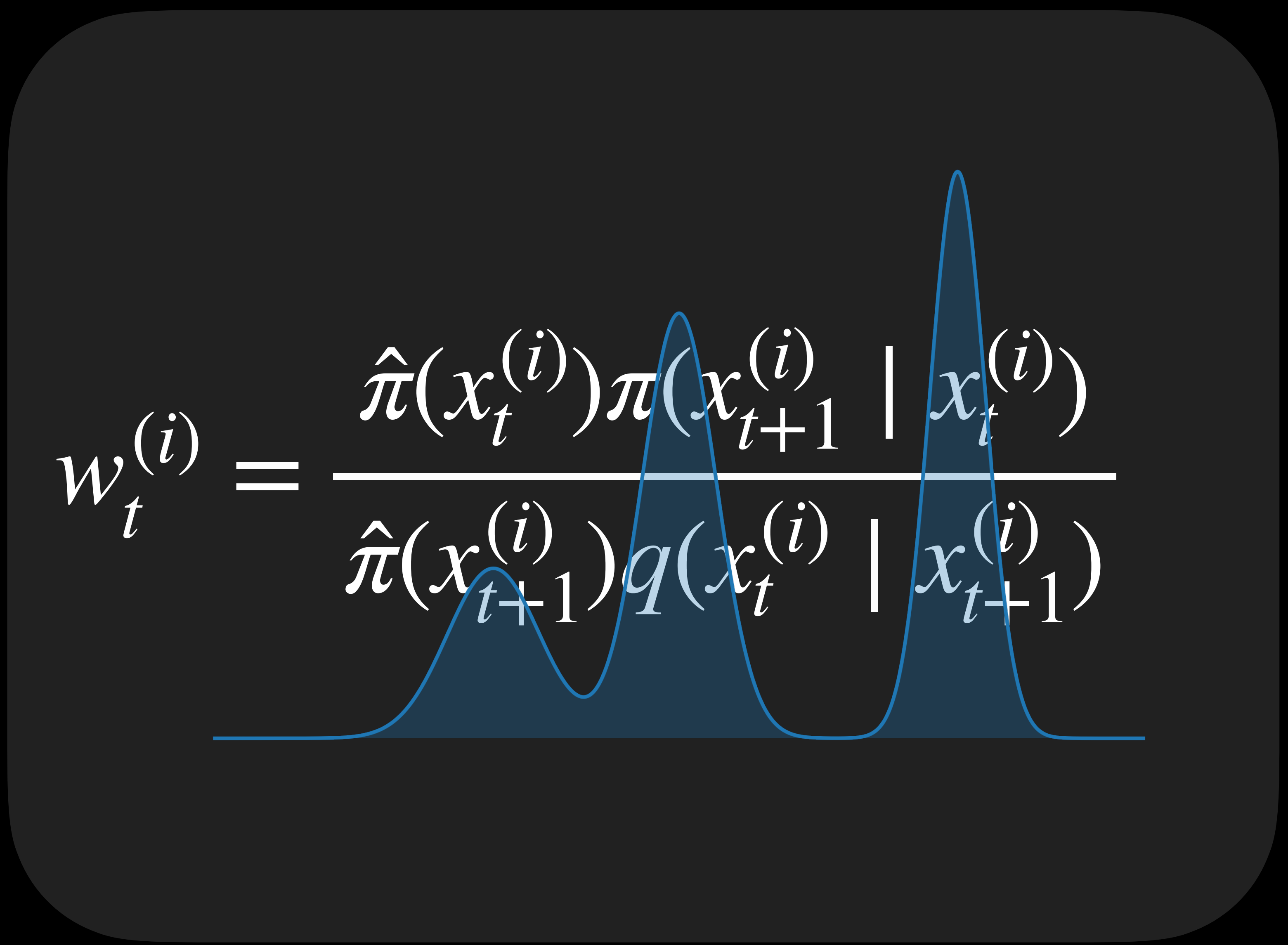
Can we develop a sampler based on reverse diffusion dynamics?

Reverse Diffusion Sequential Monte Carlo (RDSMC)

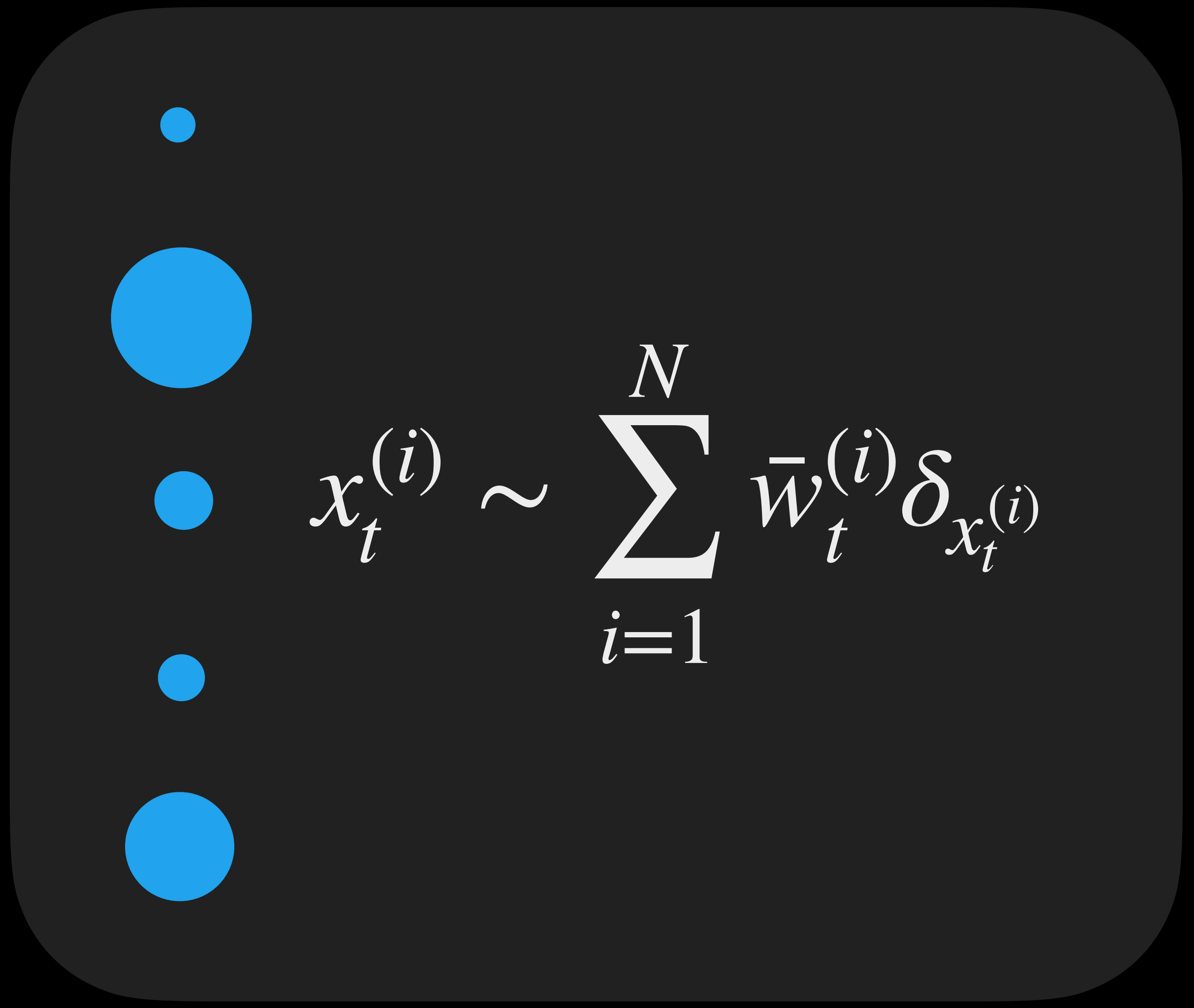
$$dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$$



Diffusion proposal
via score estimates



Weighting via
marginal estimates

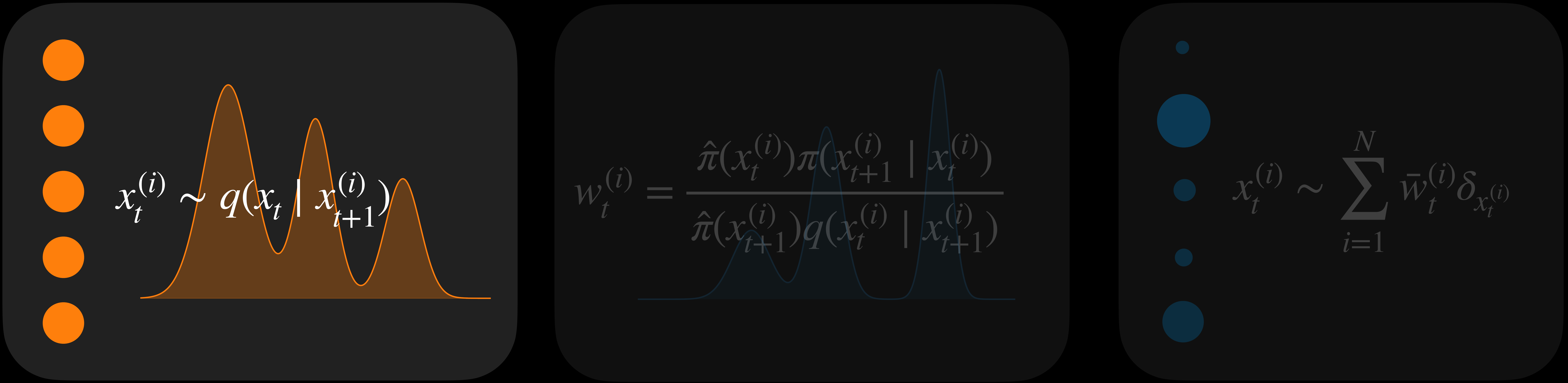


Resampling

$\times$ *T steps*

Reverse Diffusion Sequential Monte Carlo (RDSMC)

$$dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$$



Diffusion proposal
via score estimates

Diffusion proposal via score estimates

$$dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$$

- Score is intractable; use Monte Carlo approximation

$$\nabla_{x_t} \log \pi(x_t) = \int \frac{\alpha_t x_0 - x_t}{\sigma_t^2} \pi(x_0 \mid x_t) dx_0 \approx s(x_t) = \sum_{m=1}^{M} \bar{w}^{(m)} \frac{\alpha_t x_0^{(m)} - x_t}{\sigma_t^2}$$

$\{x_0^{(m)}; w^{(m)}\}_m \sim \pi(x_0 \mid x_t)$

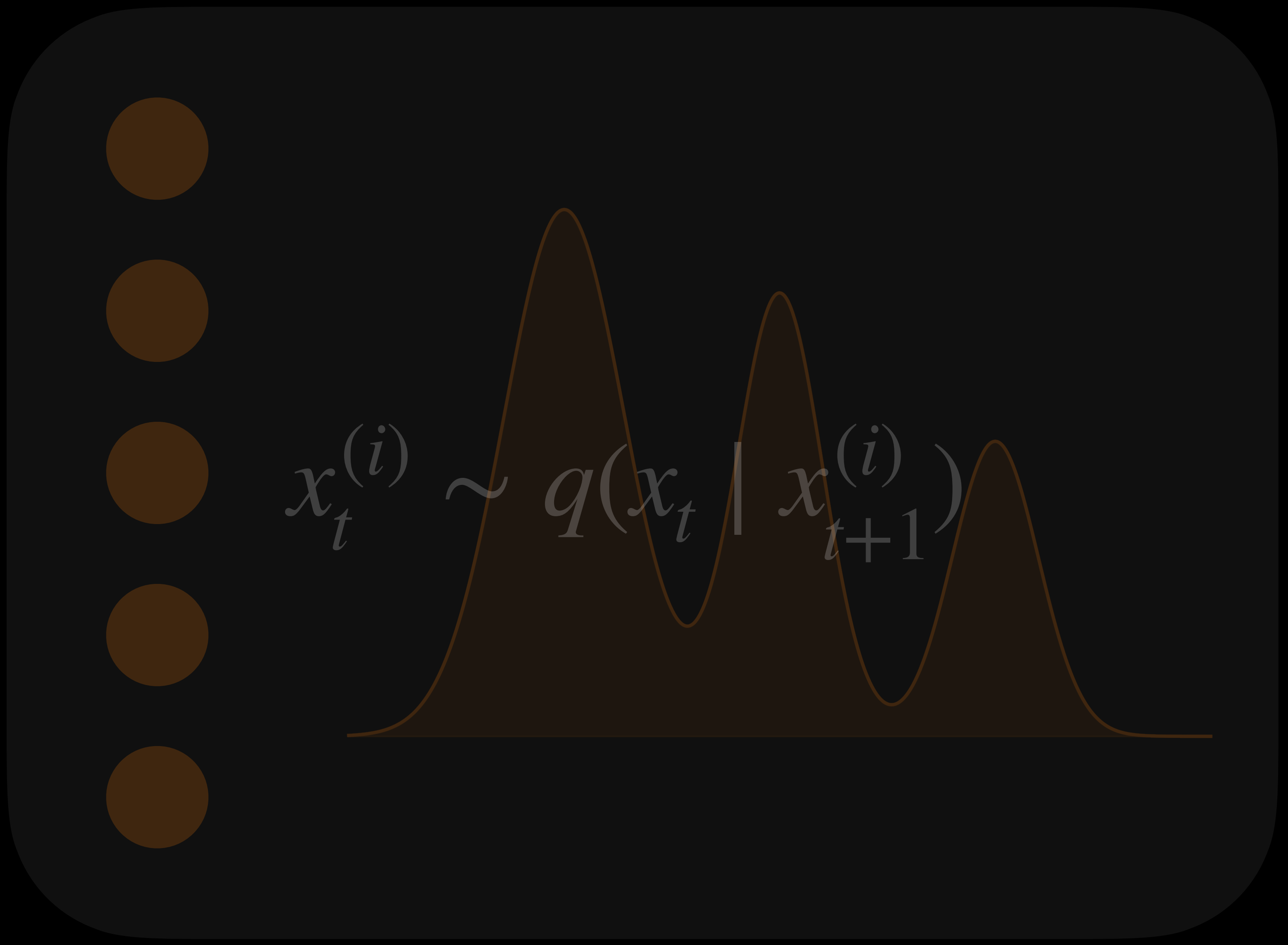
importance sampling or AIS

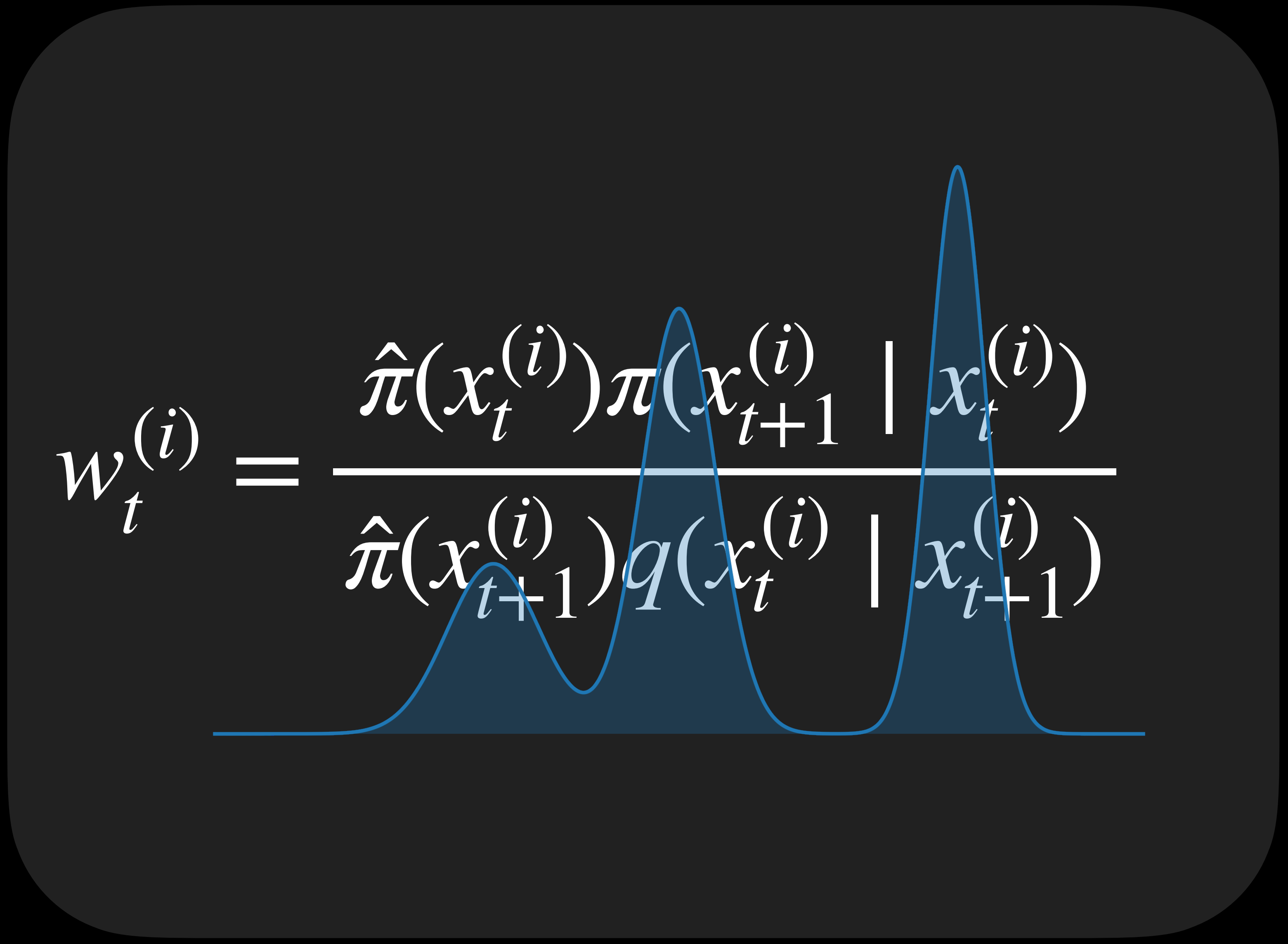
- Proposal

$$q(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1} \mid x_t - [f_t x_t - g_t^2\, s(x_t)\,]\delta, g_t^2 \delta)$$

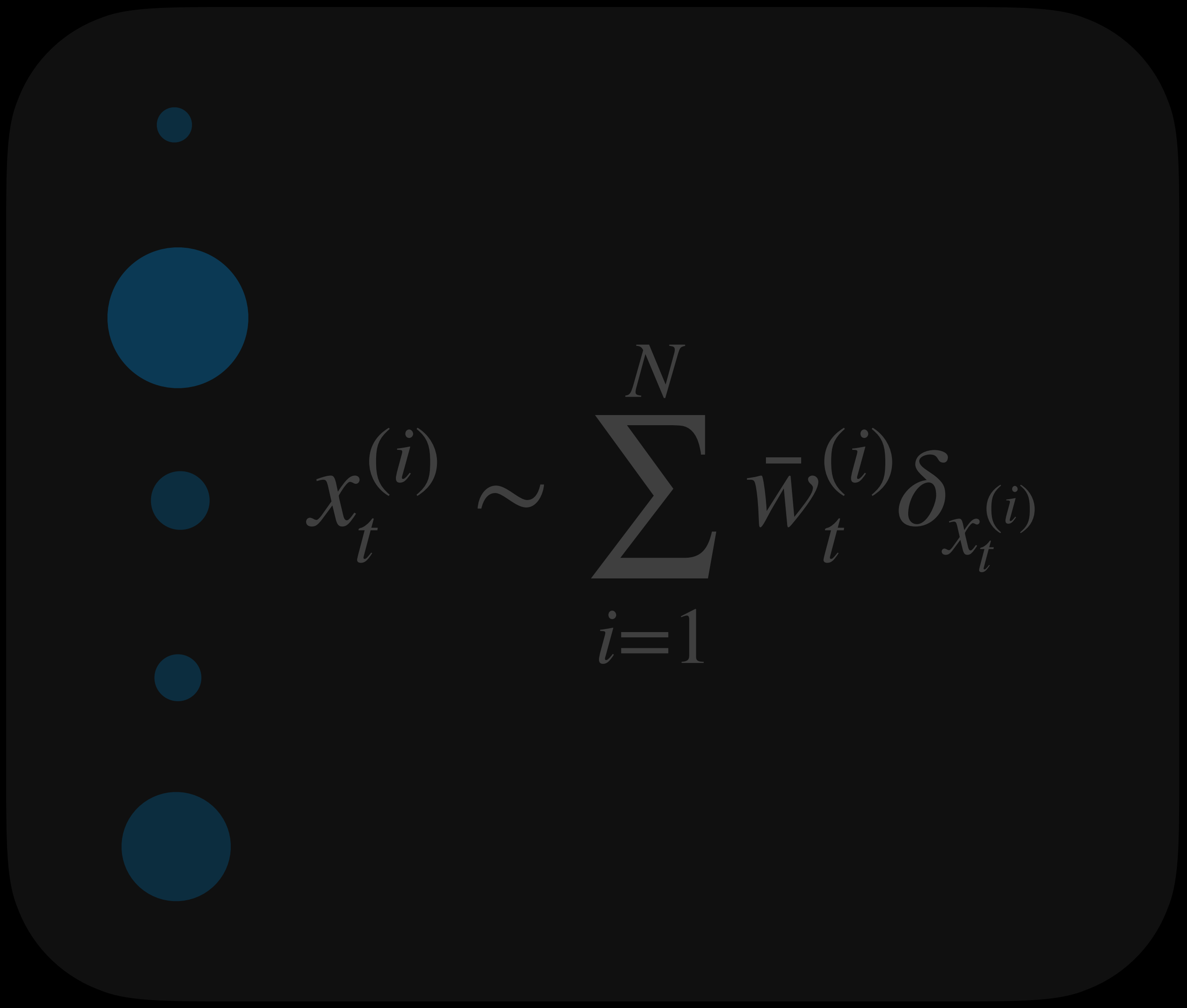
Reverse Diffusion Sequential Monte Carlo (RDSMC)

$$dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$$





Weighting via
marginal estimates



Weighting via marginal estimates

- Use intermediate targets to align the proposed samples

$$\pi(x_t) = \int \pi(x_0)\pi(x_t \mid x_0)dx_0 \quad \approx \hat{\pi}(x_t) \ := \frac{1}{M}\sum_{m=1}^{M} w^{(m)}$$

$\{x_0^{(m)}; w^{(m)}\}_m \sim \pi(x_0 \mid x_t)$

importance sampling or AIS

❌ $\pi(x_t)$ is intractable. 💡 $\pi(x_t)$ is normalization constant of $\pi(x_0 \mid x_t)$

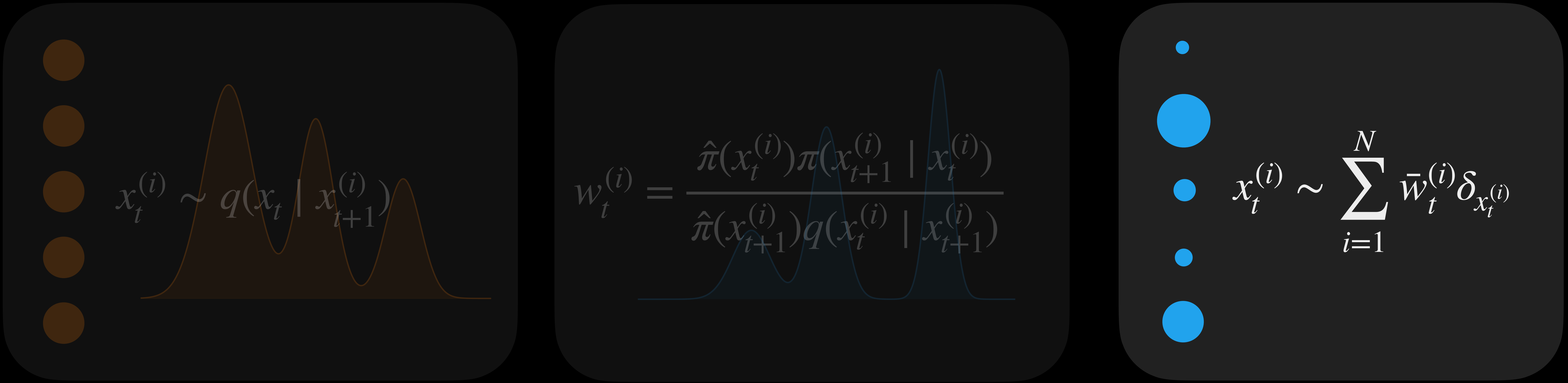
✅ Marginal estimates as byproducts of score estimation

- Develop intermediate weights to correct proposal bias

$$w_t = \frac{\hat{\pi}(x_t)\pi(x_{t+1} \mid x_t)}{\hat{\pi}(x_{t+1})q(x_t \mid x_{t+1})}$$

Reverse Diffusion Sequential Monte Carlo (RDSMC)

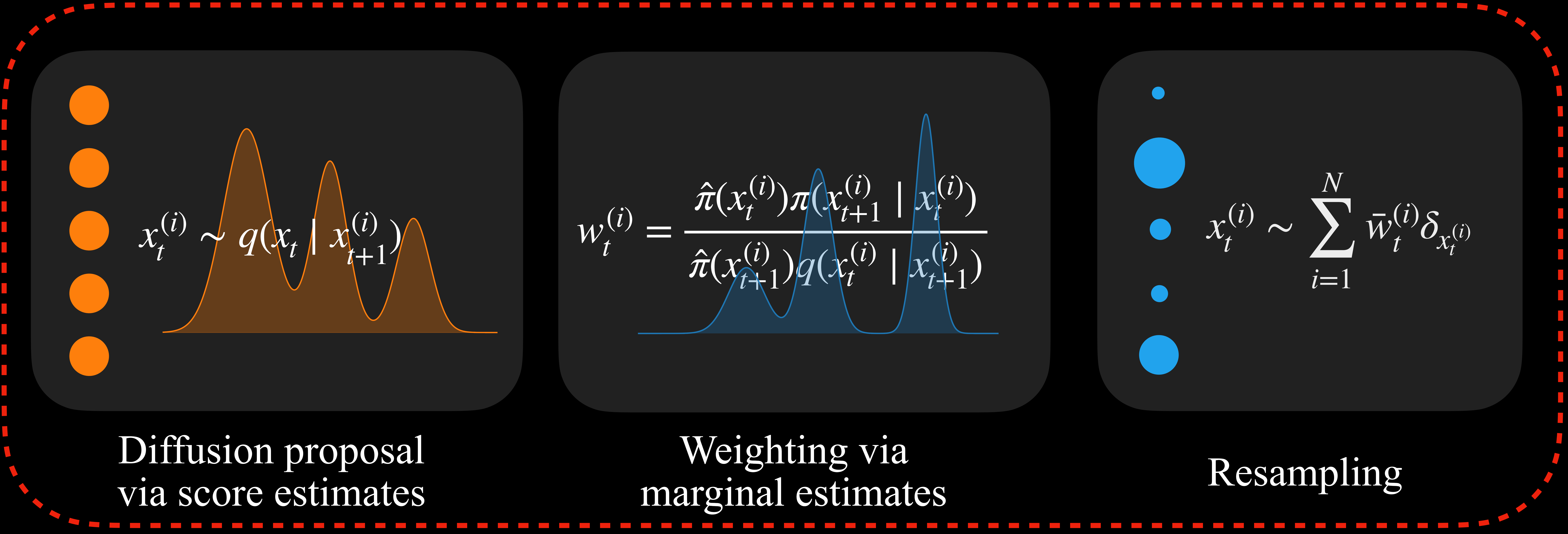
$$dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$$



Resampling

Reverse Diffusion Sequential Monte Carlo (RDSMC)

$$dx_t = f_t x_t - g_t^2 \nabla_{x_t} \log \pi(x_t)\, dt + g_t dB_t$$



$\times$ *T steps*

Theoretical guarantees

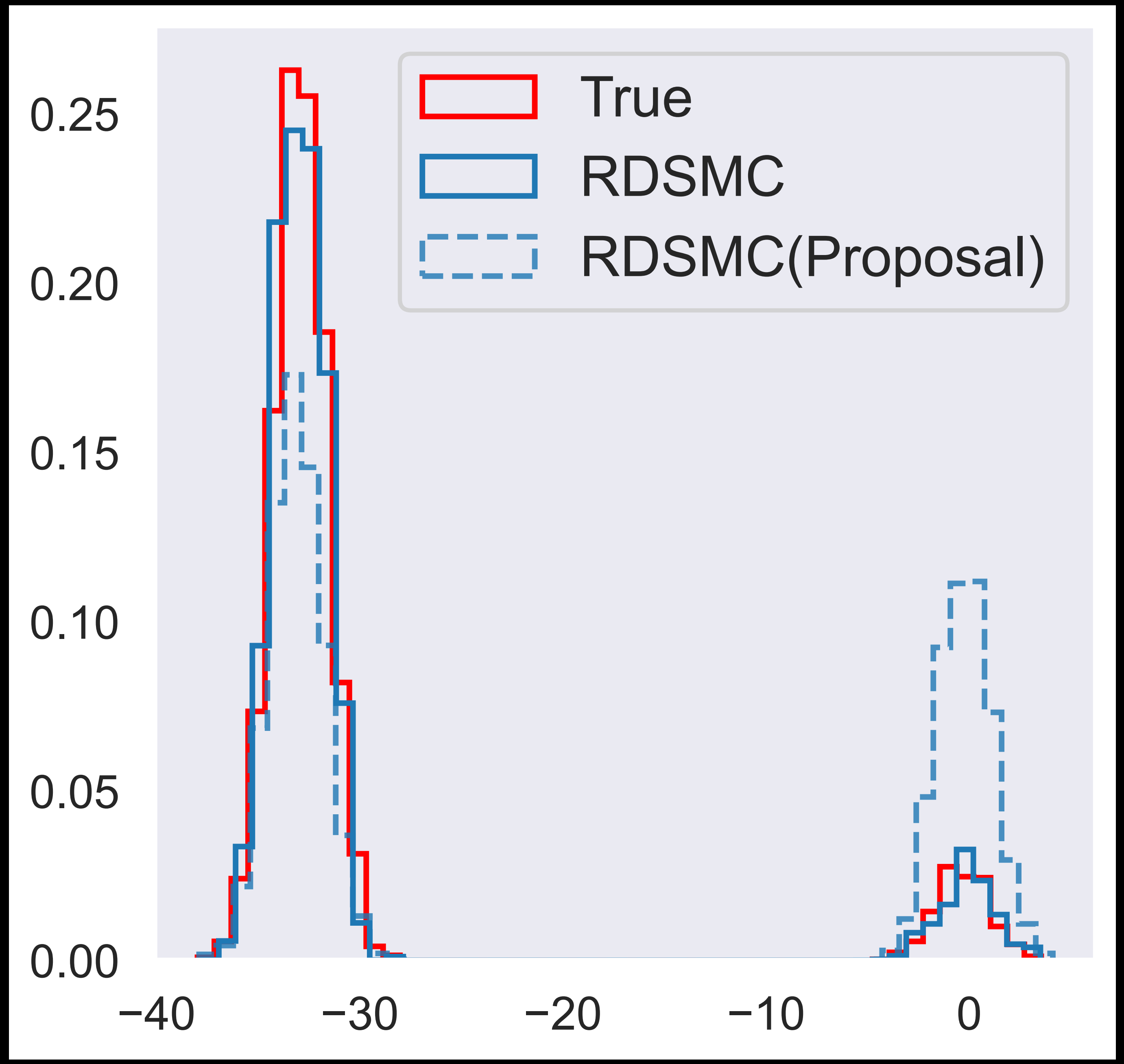
💡 RDSMC is a class of *nested SMC* methods [Naesseth et. al. 2015]

➡ Each SMC step involves Monte Carlo estimation

✅ RDSMC produces *asymptotically exact samples* from the target as the number of particles $N \to \infty$

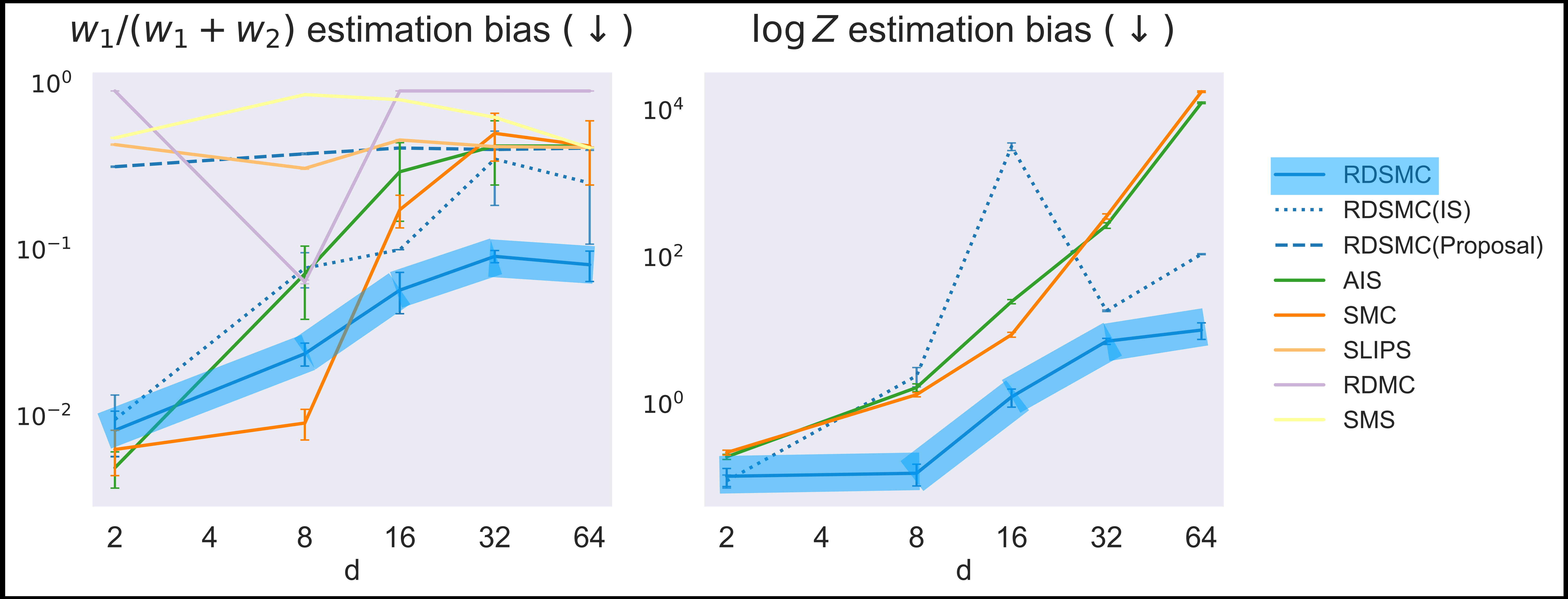
✅ RDSMC provides an *unbiased estimate of normalization constant* for any fixed N

Results: GMM



RDSMC produces calibrated weight for both modes in GMM.

Results: GMM



RDSMC has consistently lower estimation bias across dimensions.

Results: other targets

Algorithm	Rings ($d = 2$)		Funnel ($d = 10$)	
	Radius TVD (↓)	$\log Z$ Bias (↓)	Sliced KSD (↓)	$\log Z$ Bias (↓)
AIS	0.10 ± 0.00	0.05 ± 0.00	0.07 ± 0.00	**0.28 ± 0.01**
SMC	0.10 ± 0.00	0.05 ± 0.00	0.07 ± 0.00	**0.28 ± 0.01**
SMS	0.24 ± 0.01	N/A	0.15 ± 0.00	N/A
RDMC	0.37 ± 0.00	N/A	0.13 ± 0.00	N/A
SLIPS	0.19 ± 0.00	N/A	**0.06 ± 0.00**	N/A
RDSMC	0.13 ± 0.01	**0.03 ± 0.00**	0.11 ± 0.00	**0.28 ± 0.10**
RDSMC(IS)	0.15 ± 0.01	**0.02 ± 0.01**	0.33 ± 0.03	1.61 ± 0.14
RDSMC(Proposal)	**0.09 ± 0.00**	N/A	0.32 ± 0.03	N/A

Test LL (↑)	Credit ($d = 25$)	Cancer ($d = 31$)	Ionosphere ($d = 35$)	Sonar ($d = 61$)
AIS	$\mathbf{-122.73 \pm 0.51}$	-60.45 ± 0.31	$\mathbf{-86.37 \pm 0.10}$	-110.11 ± 0.06
SMC	-123.17 ± 0.05	-60.28 ± 0.11	$\mathbf{-86.37 \pm 0.10}$	-110.11 ± 0.06
SMS	-527.79 ± 0.85	-215.64 ± 0.66	-202.56 ± 0.16	-275.44 ± 0.31
RDMC	-388.24 ± 1.75	-182.81 ± 0.43	-108.67 ± 0.09	-128.29 ± 0.03
SLIPS	$\mathbf{-121.79 \pm 0.04}$	$\mathbf{-56.26 \pm 0.08}$	$\mathbf{-85.07 \pm 0.07}$	$\mathbf{-102.39 \pm 0.03}$
RDSMC	$\mathbf{-124.00 \pm 1.96}$	-62.23 ± 1.96	$\mathbf{-87.72 \pm 1.75}$	$\mathbf{-101.52 \pm 1.84}$
RDSMC(IS)	-144.38 ± 4.63	-82.47 ± 7.78	$\mathbf{-84.90 \pm 2.84}$	$\mathbf{-110.57 \pm 4.54}$
RDSMC(Proposal)	-606.03 ± 1.26	-246.86 ± 0.97	-92.62 ± 0.04	-134.72 ± 0.13

RDSMC outperforms or are comparable to other baselines

Check out our paper and code for more details!

Paper:

https://arxiv.org/abs/2508.05926

Code:

https://github.com/LuhuanWu/RDSMC