Monte Carlo Variational Inference

Justin Domke, Computer Science, University of Massachusetts Amherst

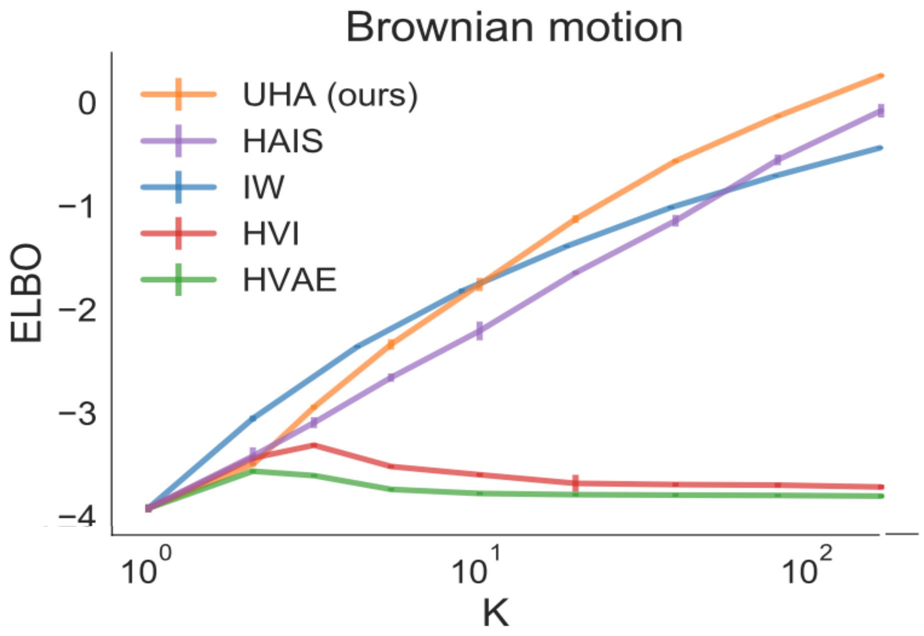
Sample state $z_1 \sim q$ and momentum $\rho_1 \sim S$.

Initialize estimator as $\mathcal{L} \leftarrow -\log q(z_1)$.

For $m = 1,2,\cdots,K-1$:

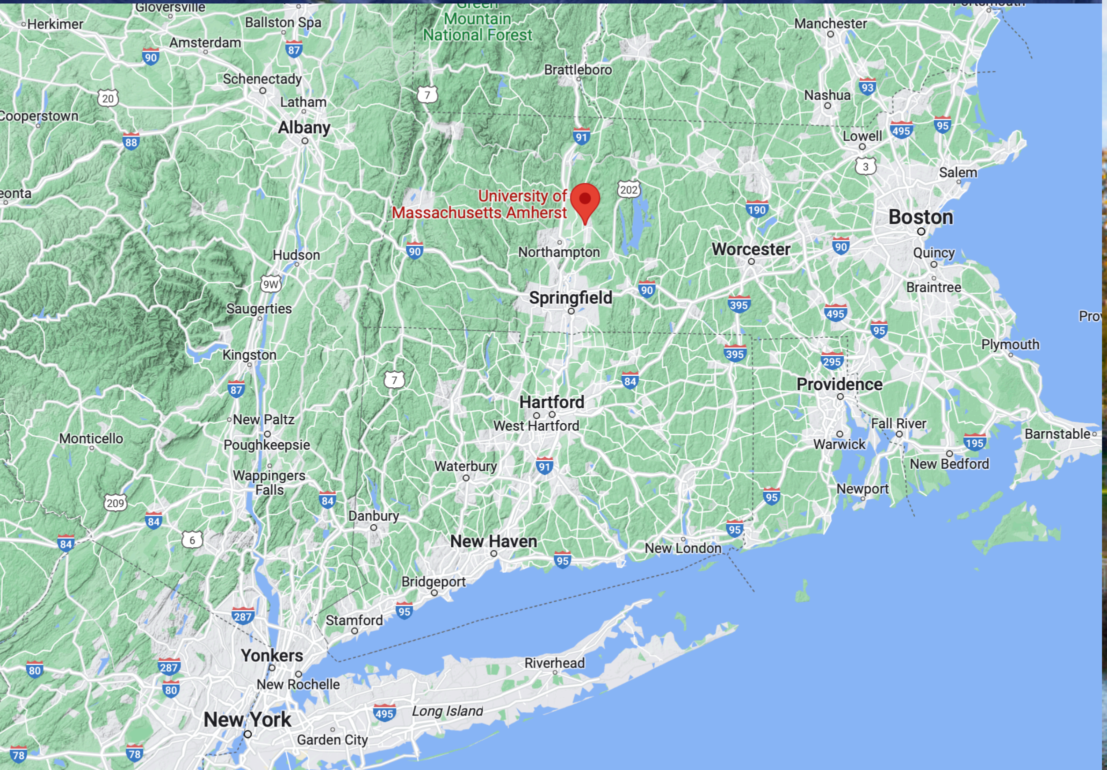
- Sample new momentum ρ'.
- $(z_{k+1}, \rho_{k+1}) \leftarrow$ HMC with target π_k and starting point (z_k, ρ').
- Update estimator as $\mathcal{L} \leftarrow \mathcal{L} + \log S(\rho_{k+1}) - \log S(\rho')$

Update estimator as $\mathcal{L} \leftarrow \mathcal{L} + \log p(z_K, x)$



Calibration

Markov chain Monte Carlo
variational inference
sequential Monte Carlo
importance sampling
stratified sampling
KL-divergence
ELBO
Annealed importance sampling
Hamiltonian Monte Carlo

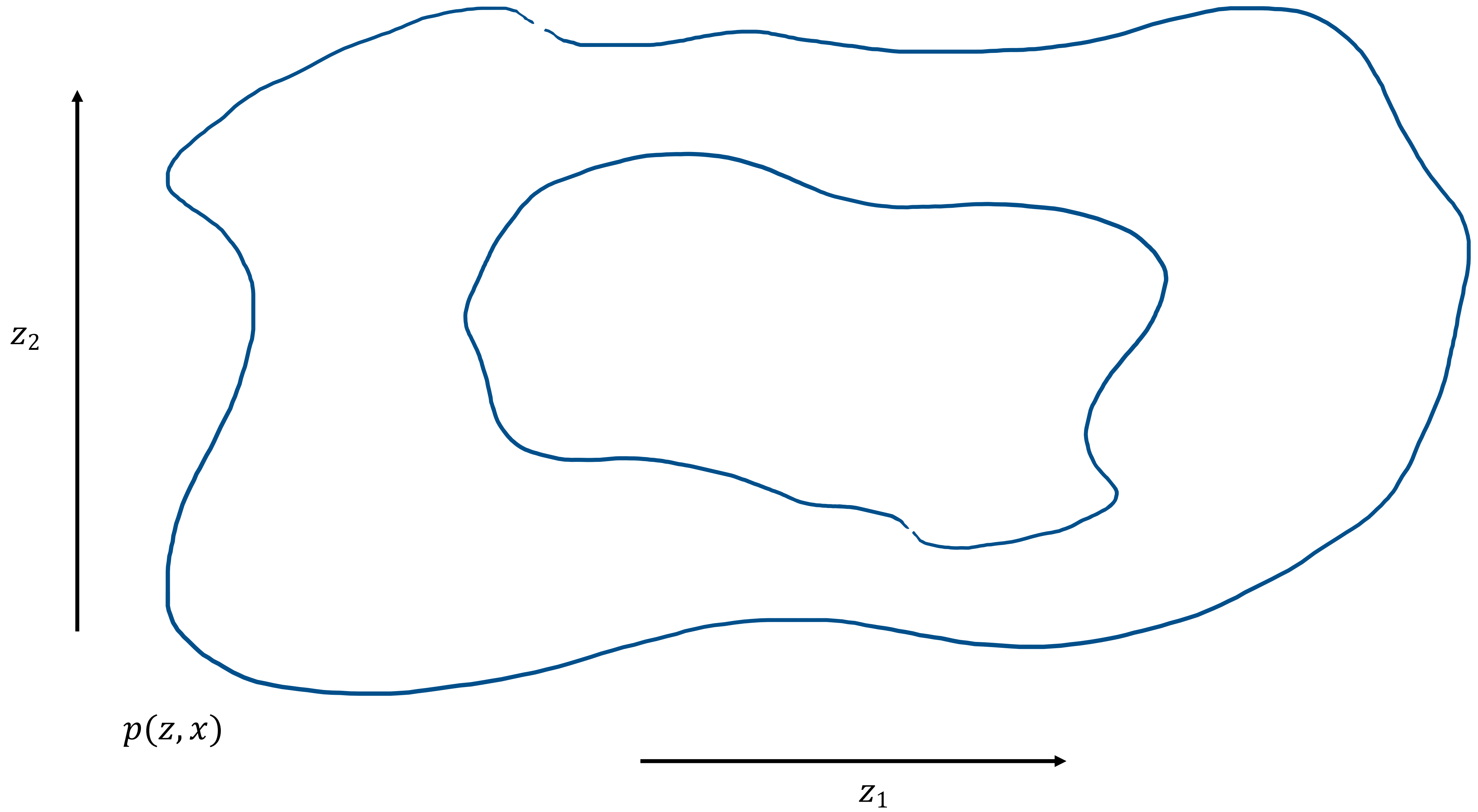

Greenland
Iceland
Finland
Sweden
Norway
Russia
United Kingdom
Canada
University of Massachusetts Amherst
Poland
Germany
Ukraine
Kazakhstan
Mongolia
France
Italy
Spain
Türkiye
China
South Korea
Japan
United States
Iraq
Iran
Afghanistan
Pakistan
Algeria
Libya
Egypt
Saudi Arabia
India
Mexico
Thailand
Mali
Niger
Chad
Sudan
Nigeria
Ethiopia
Venezuela
Colombia
DRC
Kenya
Tanzania
Indonesia
Papua New Guinea
Brazil
Peru
Bolivia
Angola
Namibia
Botswana
Madagascar
Australia
Chile
South Africa
Argentina

Gloversville
Herkimer
Ballston Spa
Green Mountain National Forest
Manchester
Amsterdam
Brattleboro
Schenectady
Latham
Albany
Nashua
Cooperstown
Lowell
Salem
University of Massachusetts Amherst
Boston
Hudson
Northampton
Worcester
Quincy
Braintree
Springfield
Saugerties
Kingston
Plymouth
Providence
Hartford
West Hartford
New Paltz
Monticello
Poughkeepsie
Fall River
Warwick
Barnstable
New Bedford
Waterbury
Wappingers Falls
Newport
Danbury
New Haven
New London
Bridgeport
Stamford
Yonkers
Riverhead
New Rochelle
New York
Long Island
Garden City


Bayesian inference

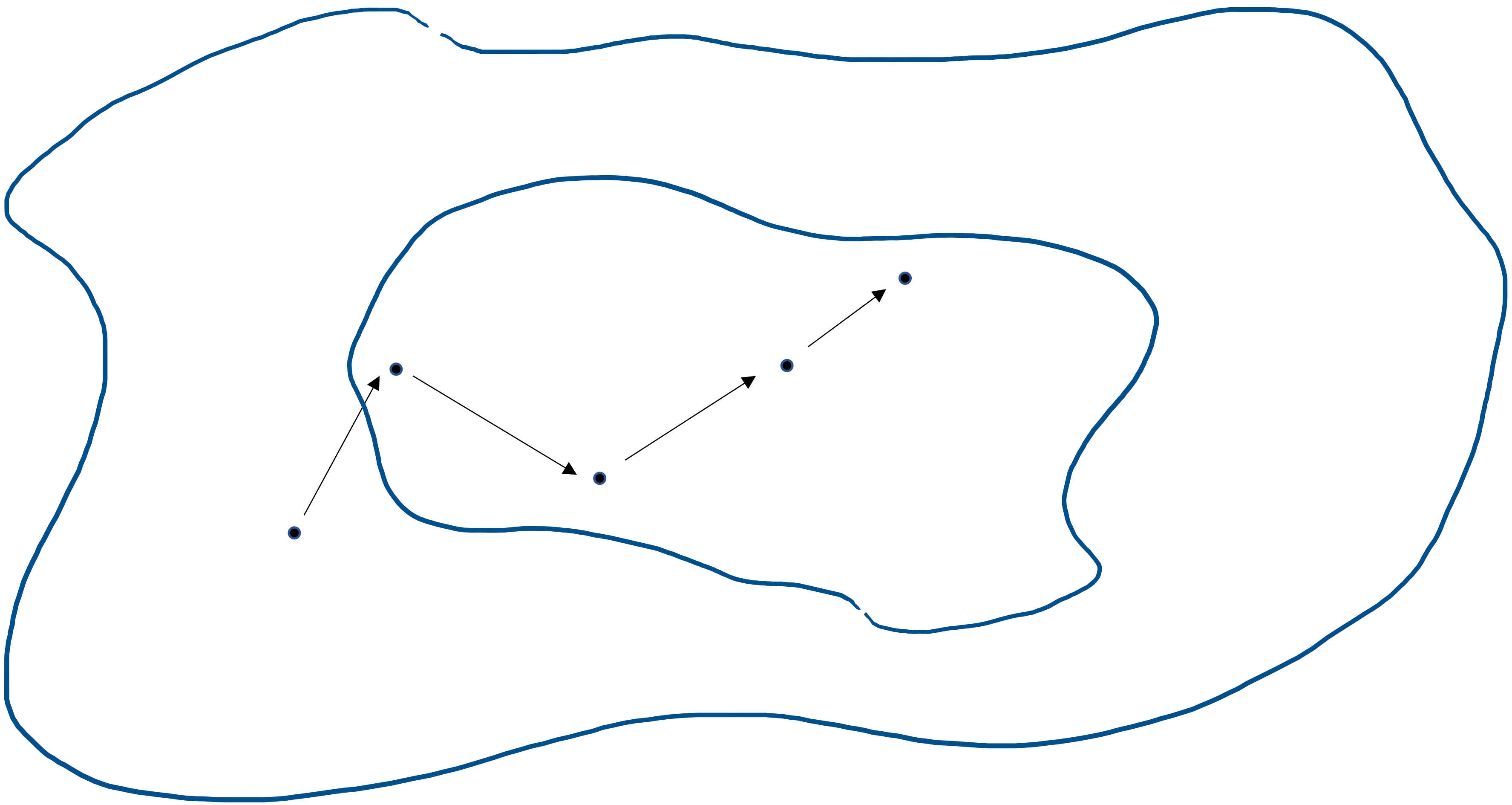
domain	**latent z**	**observed x**
epidemiology	infections, transmission rate, recovery time	cases
political science	candidate popularity, pollster biases	polls
astrophysics	14 parameters of the universe	2677 statistical measurements from galaxy survey
ecology	# animals in different locations over time	surveys
phylogenetics	evolutionary history	genomic data

1. Write down **prior** $p(z)$.
2. Write down **likelihood** $p(x|z)$.
3. Observe **data** x.
4. Use **posterior** $p(z|x)$.

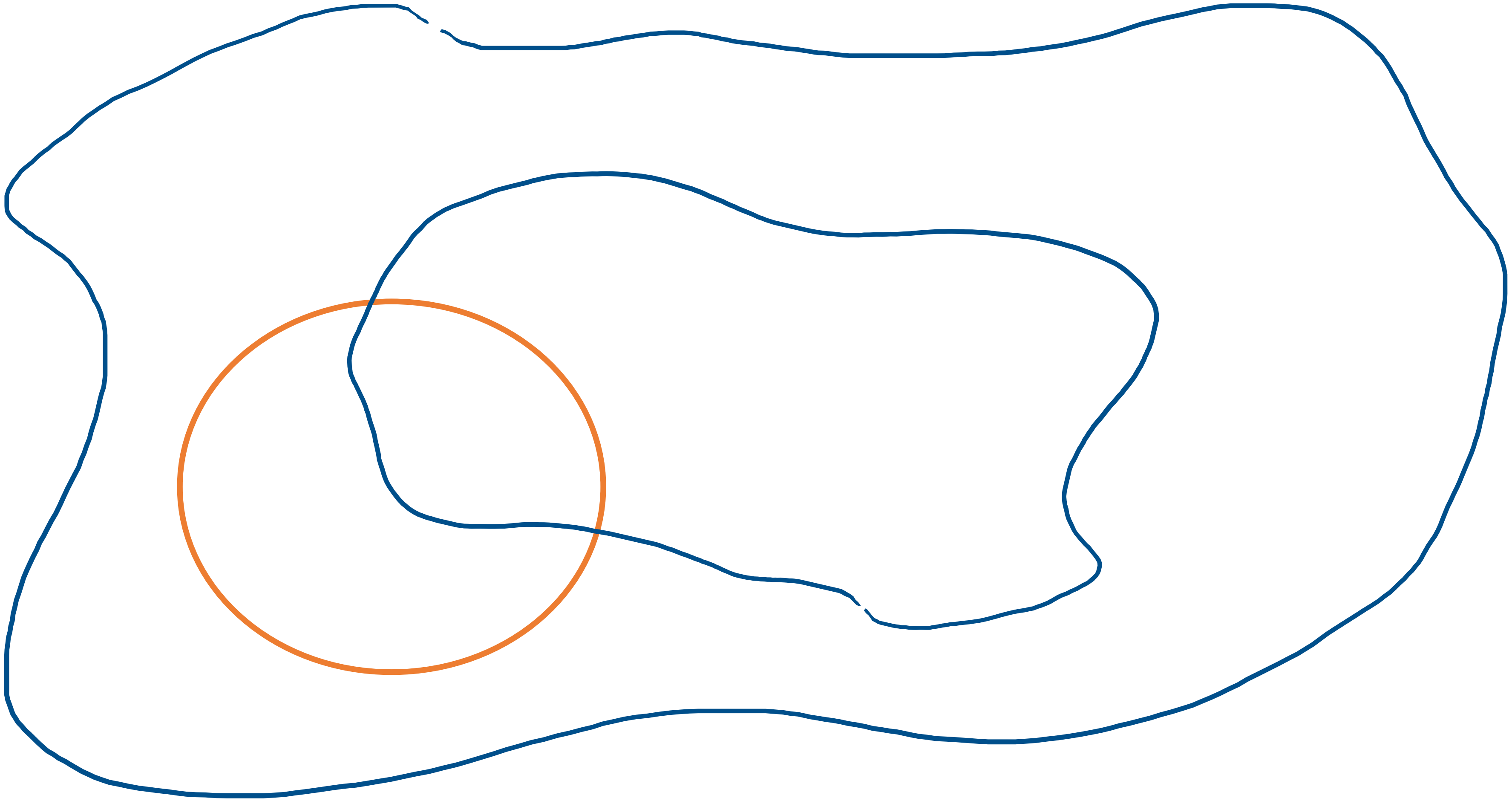
Posterior



Markov chain Monte Carlo (MCMC)

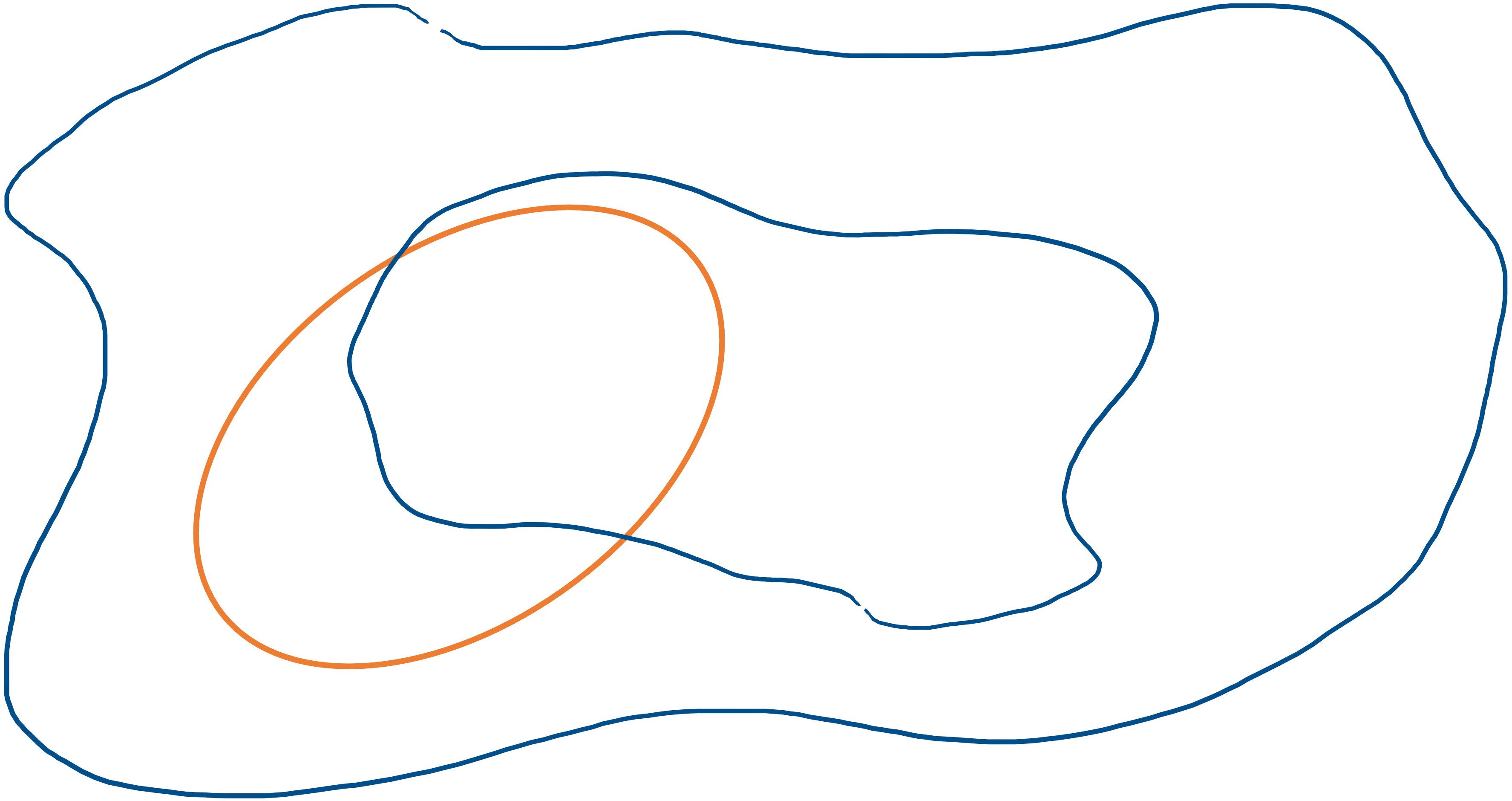


Variational Inference (VI)



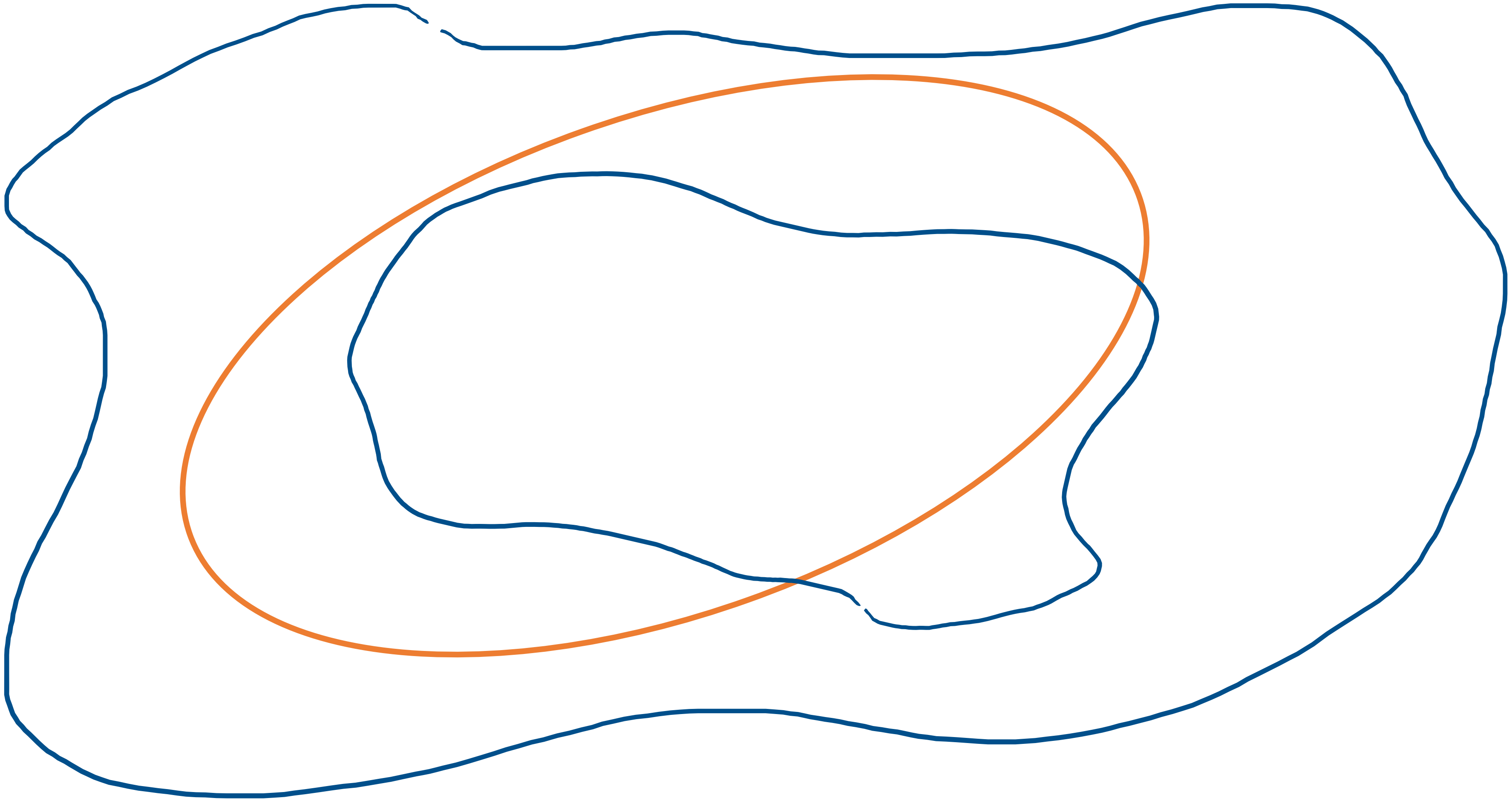
$$\min_{q \in \text{Family}} \mathrm{KL}(q \| p)$$

Variational Inference (VI)



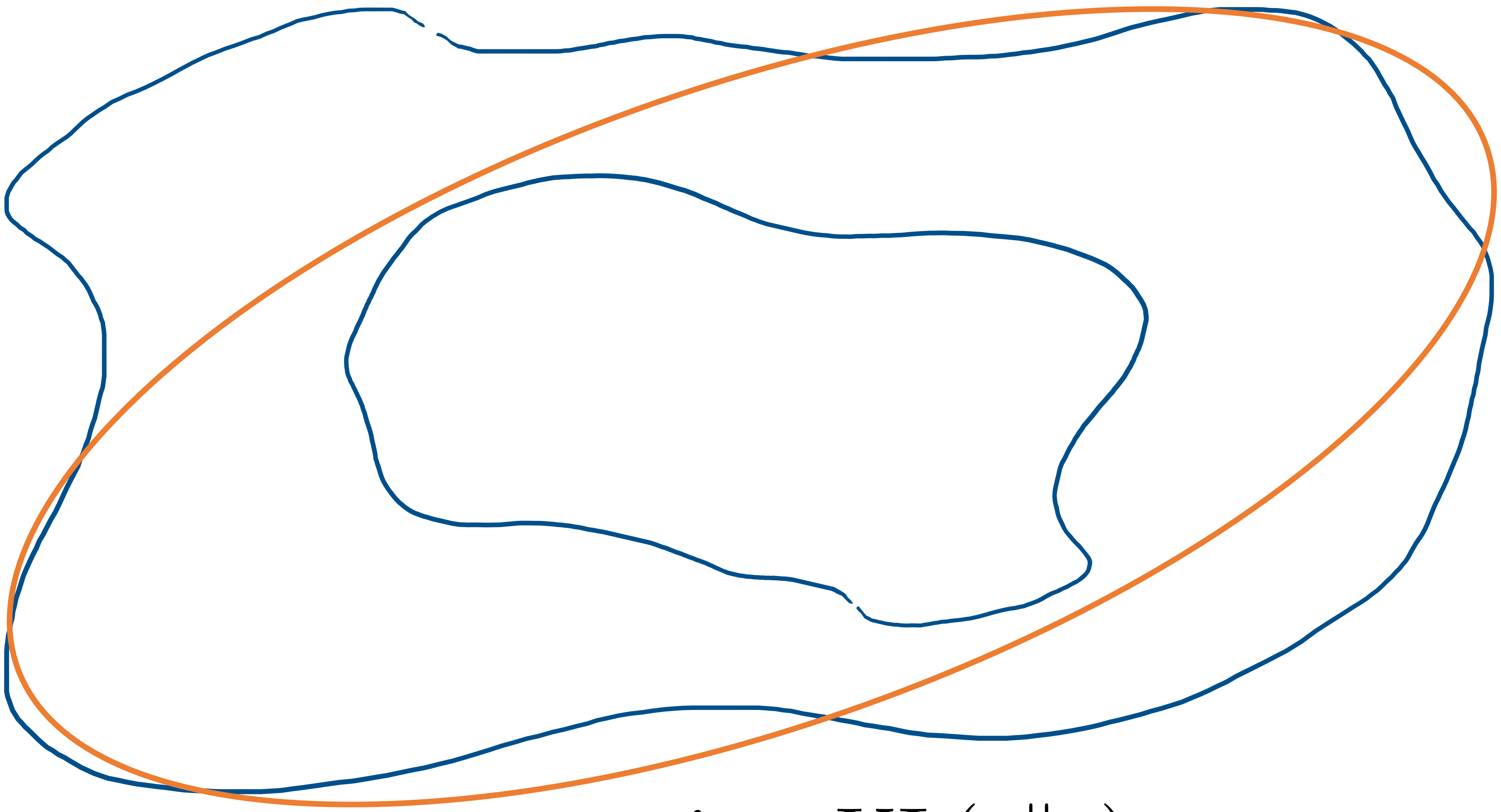
$$\min_{q \in \text{Family}} \mathrm{KL}(q \| p)$$

Variational Inference (VI)



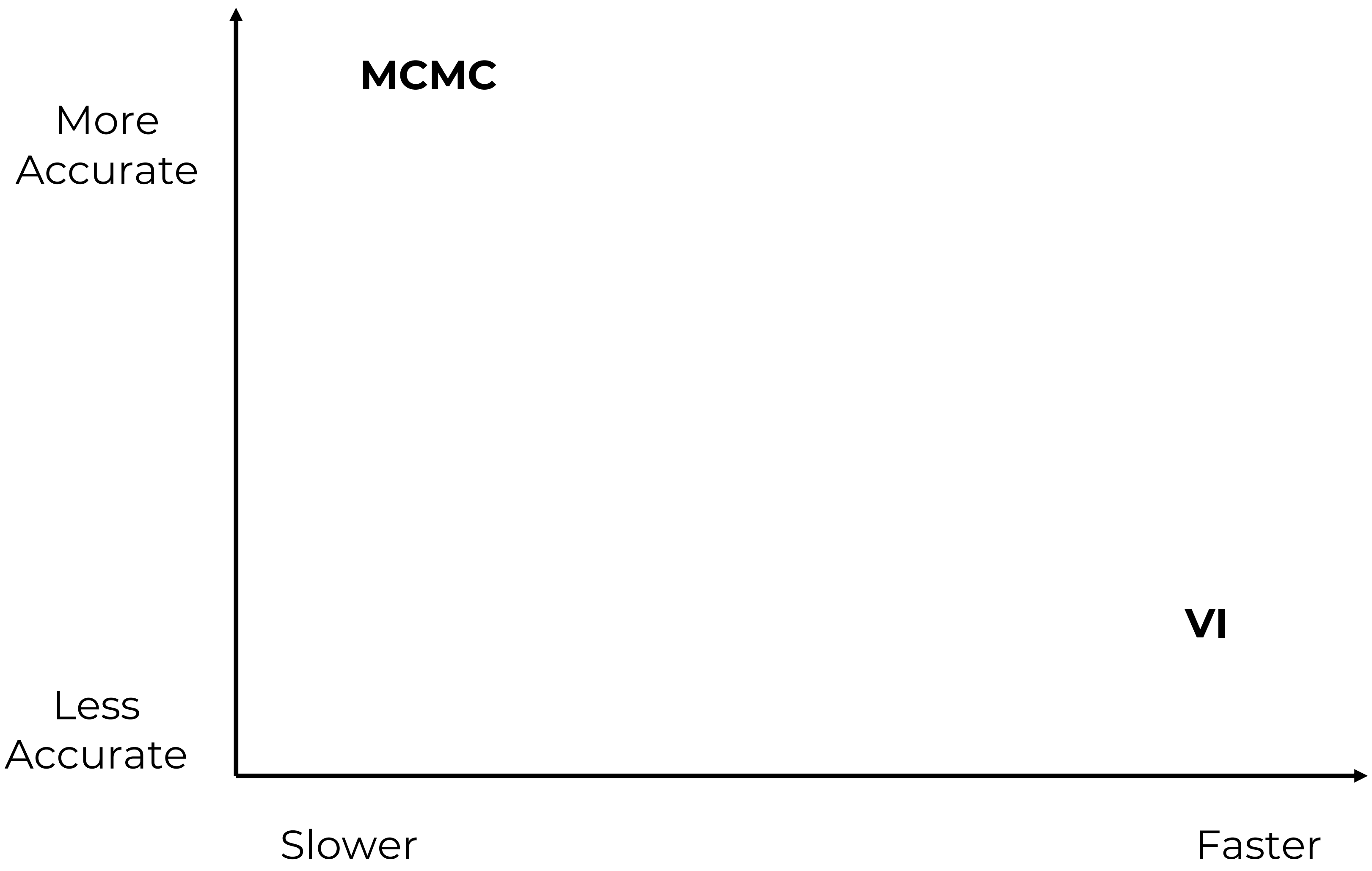
$$\min_{q \in \text{Family}} \mathrm{KL}(q \| p)$$

Variational Inference (VI)

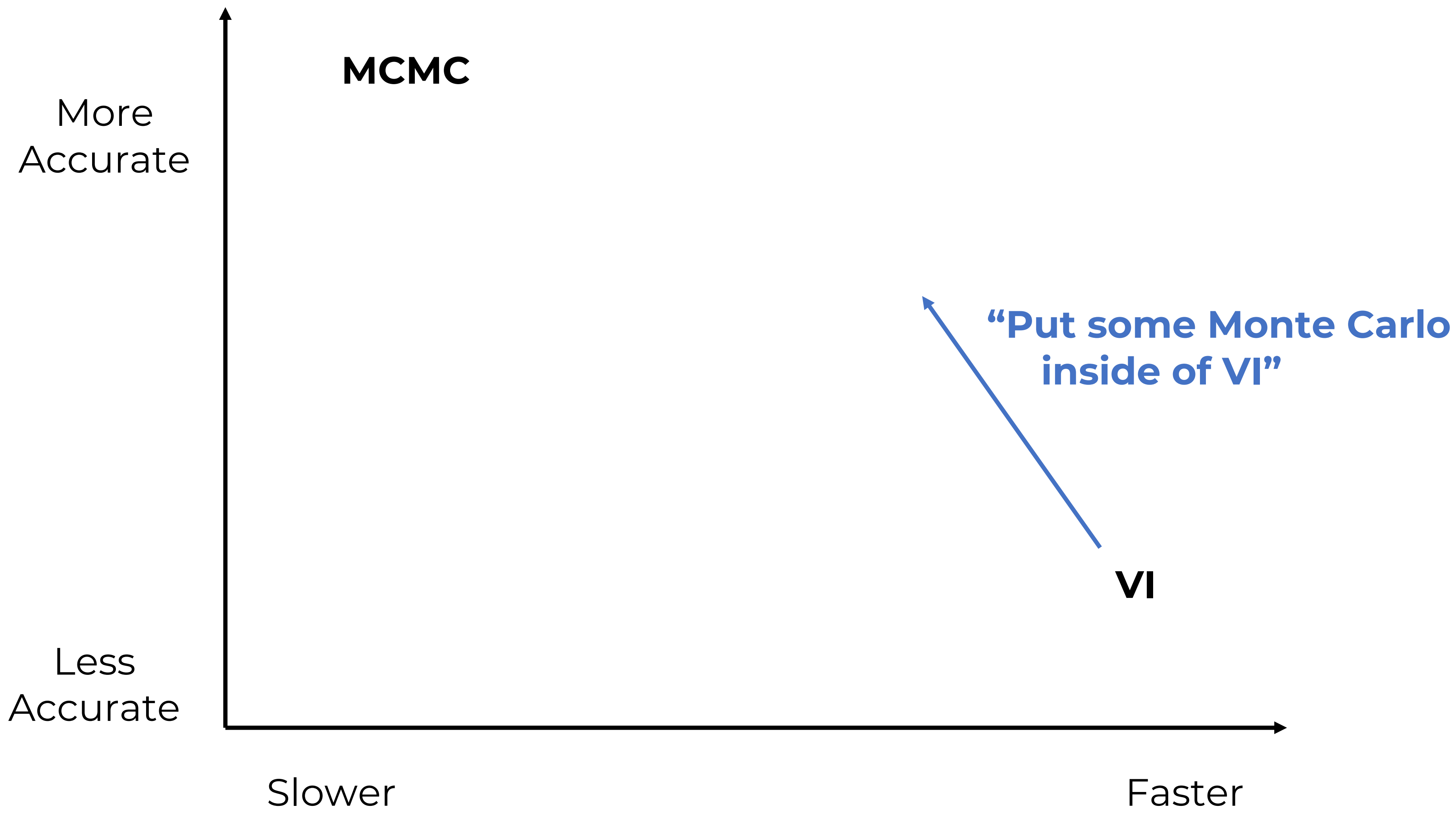


$$\min_{q \in \text{Family}} \mathrm{KL}(q \| p)$$

Folk Wisdom



This Talk


MCMC
More
Accurate
“Put some Monte Carlo
inside of VI”
VI
Less
Accurate
Slower
Faster

Outline

Importance Weighted VI

Variational Particle Filters

Variational MCMC

Importance Weighted VI

(Plus {Antithetic, Stratified, Quasi-Monte Carlo, Latin hypercube} VI)

The inference problem

Target distribution: $p(z, x)$

data: x (seen)

latent variables: z (unseen)

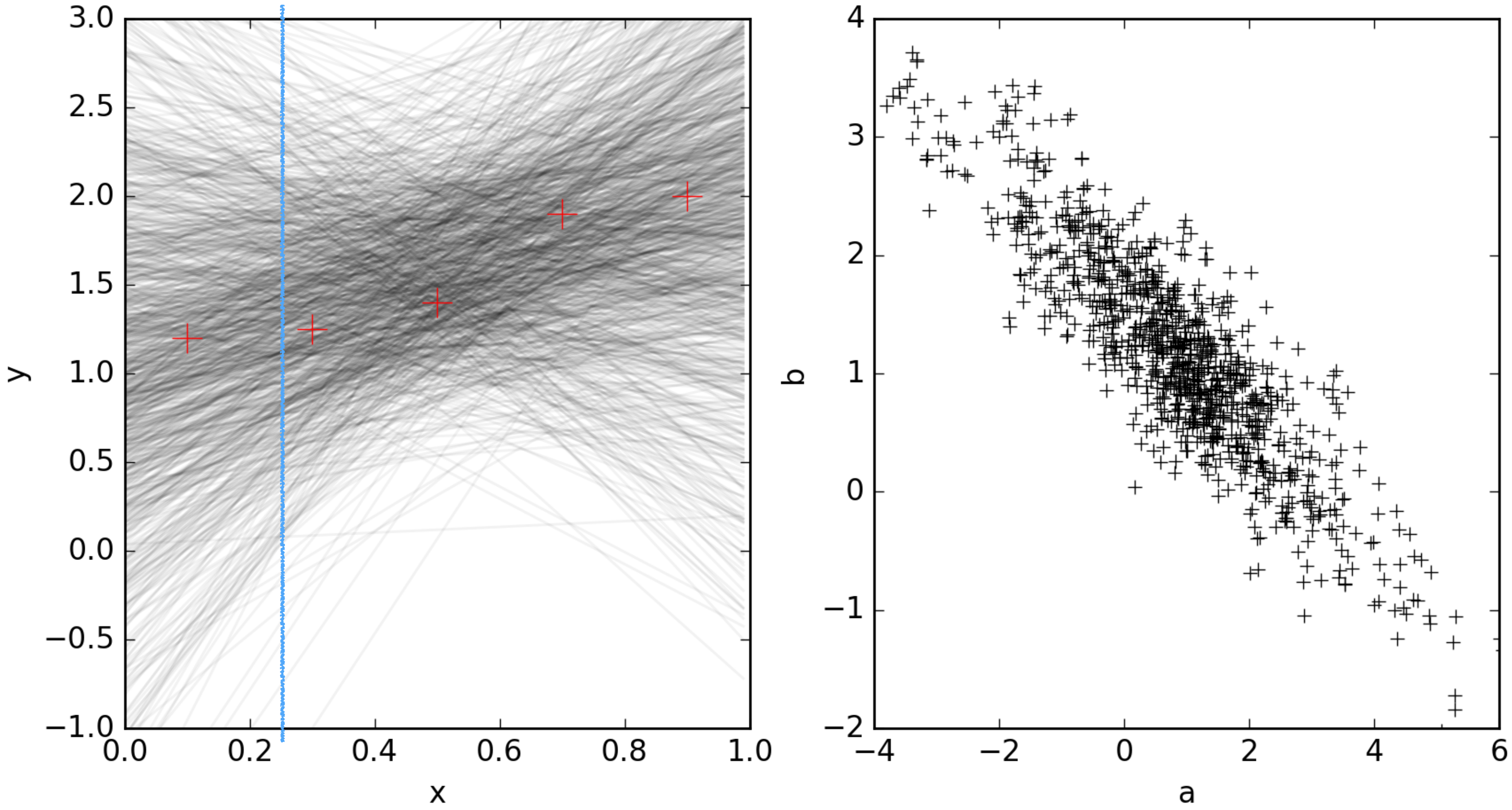
Goals:

Bayesian inference: Approximate $p(z|x)$

Learning: Maximize $\log p(x)$

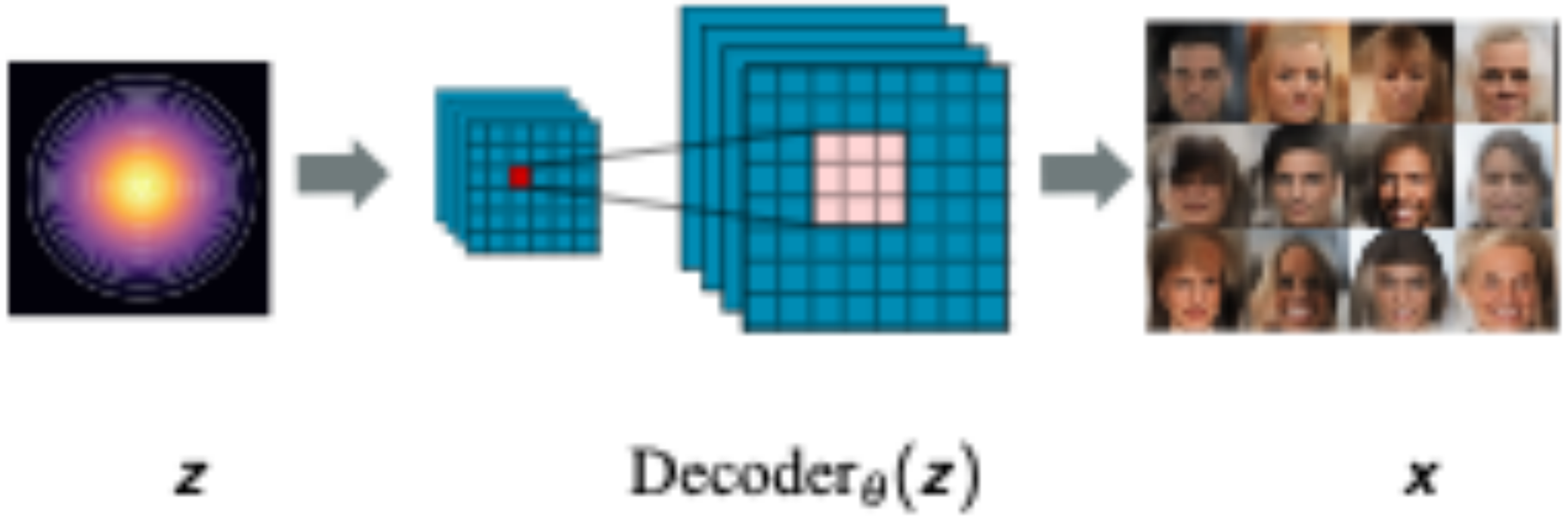
Bayesian regression

$$p(a, b|x, y) \propto p(a, b) \prod_i p(y_i|x_i, a, b)$$



Learning: Variational auto-encoder

$$p_\theta(z, x) = p(z)p_\theta(x|z)$$



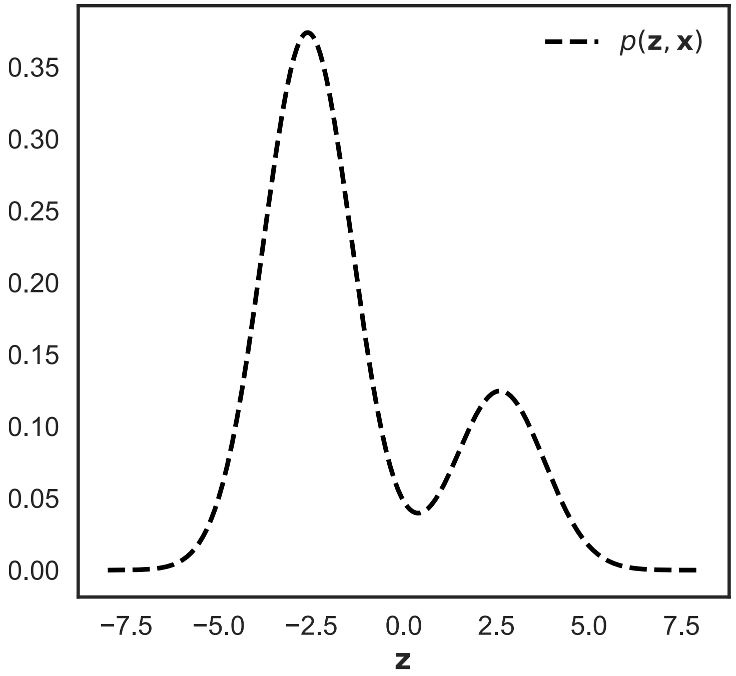
Use VI to bound $\log p_\theta(x)$

The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \underset{q(z)}{\mathbb{E}}\left[\log p(z,x) - \log q(z)\right]$$

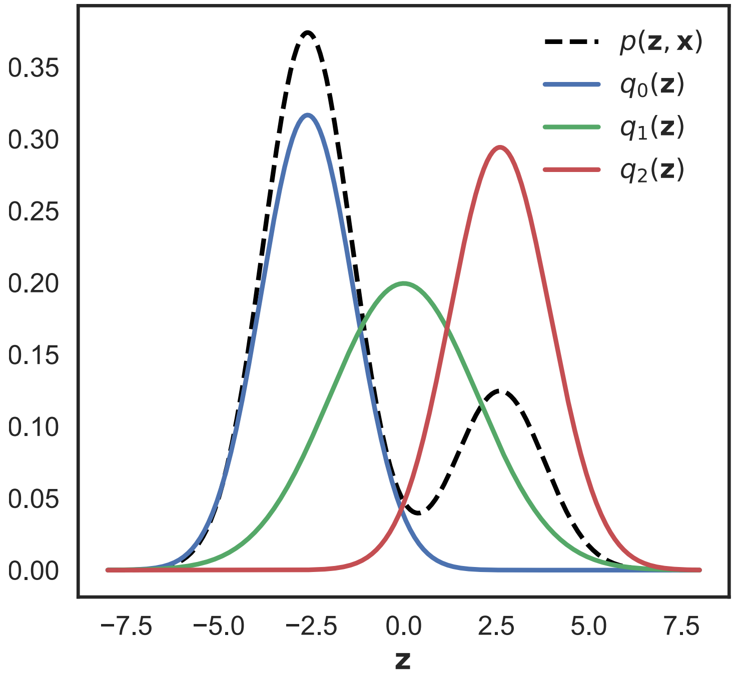


The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)} \left[\log p(z,x) - \log q(z)\right]$$

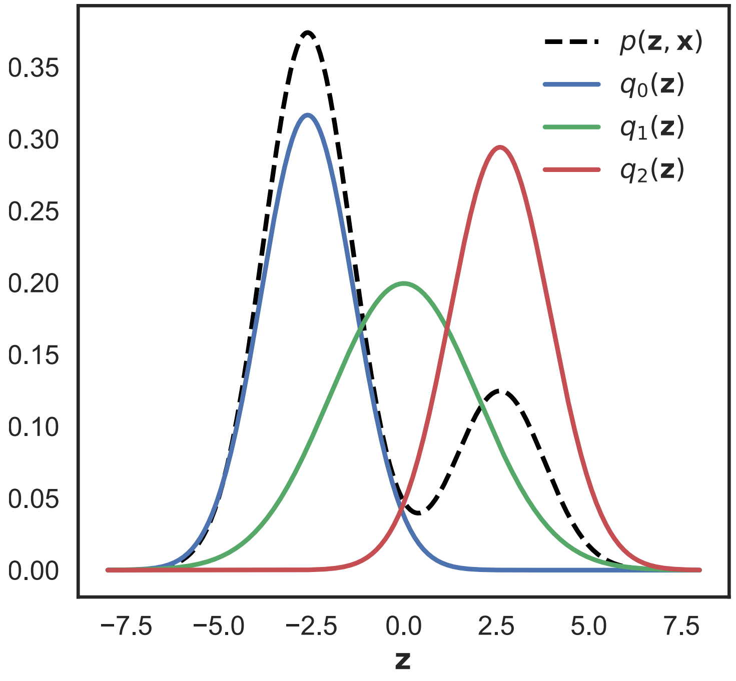


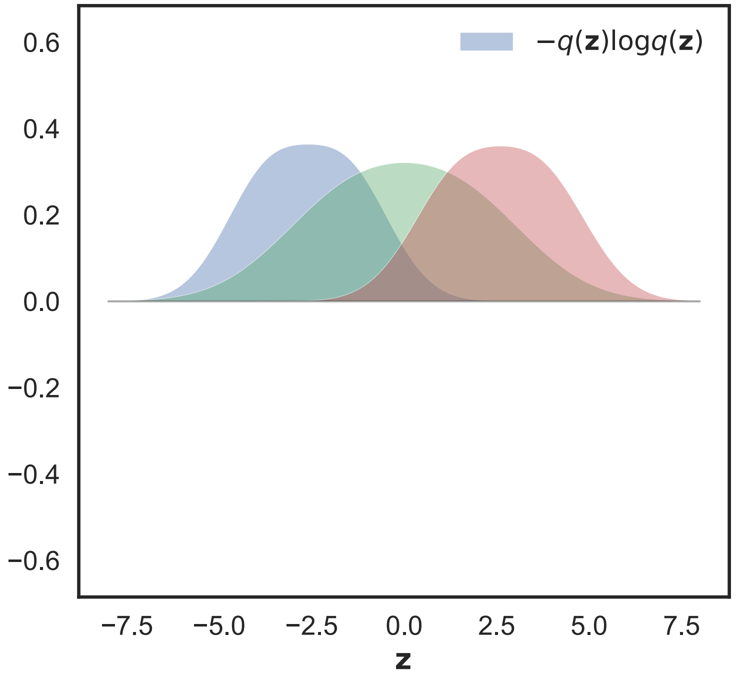
The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)}\left[\log p(z,x) - \log q(z)\right]$$



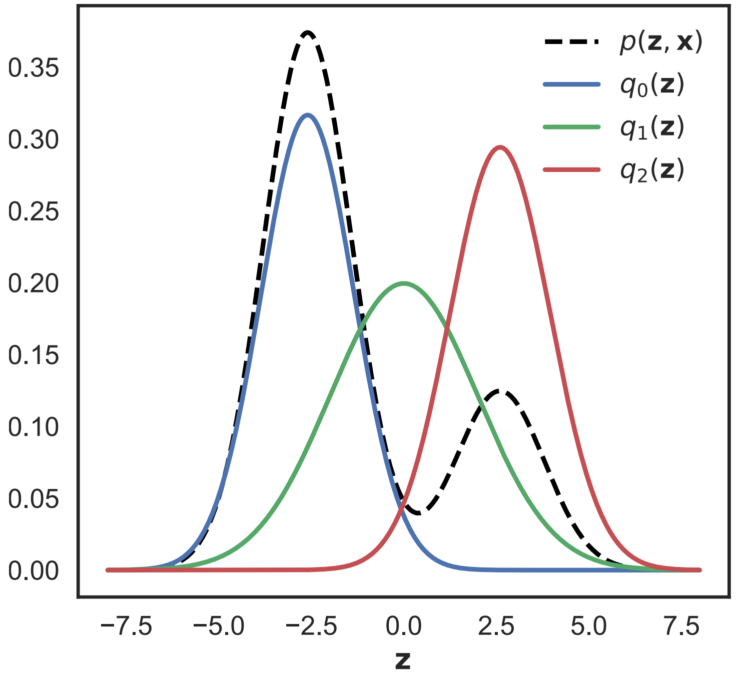


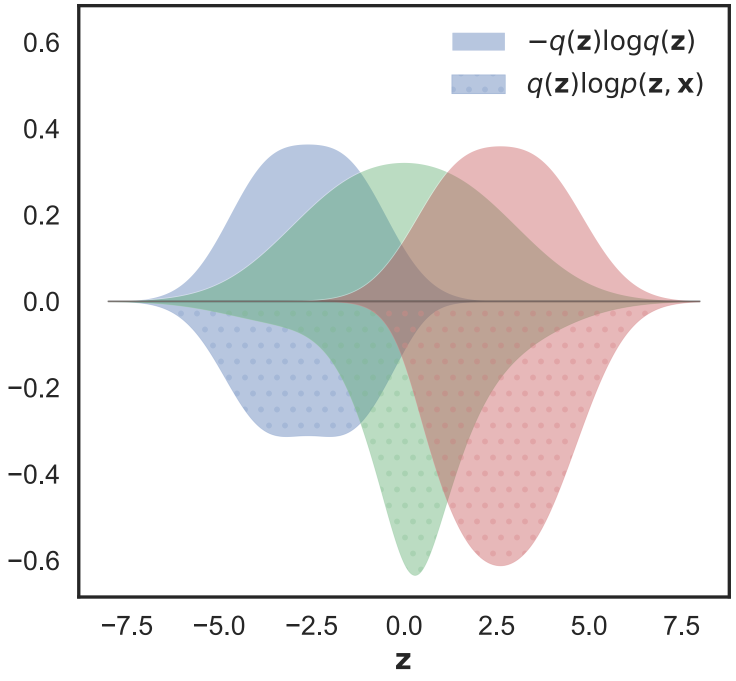
The "ELBO"

is p high?

is q spread?

$$\text{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)}\left[\log p(z,x) - \log q(z)\right]$$



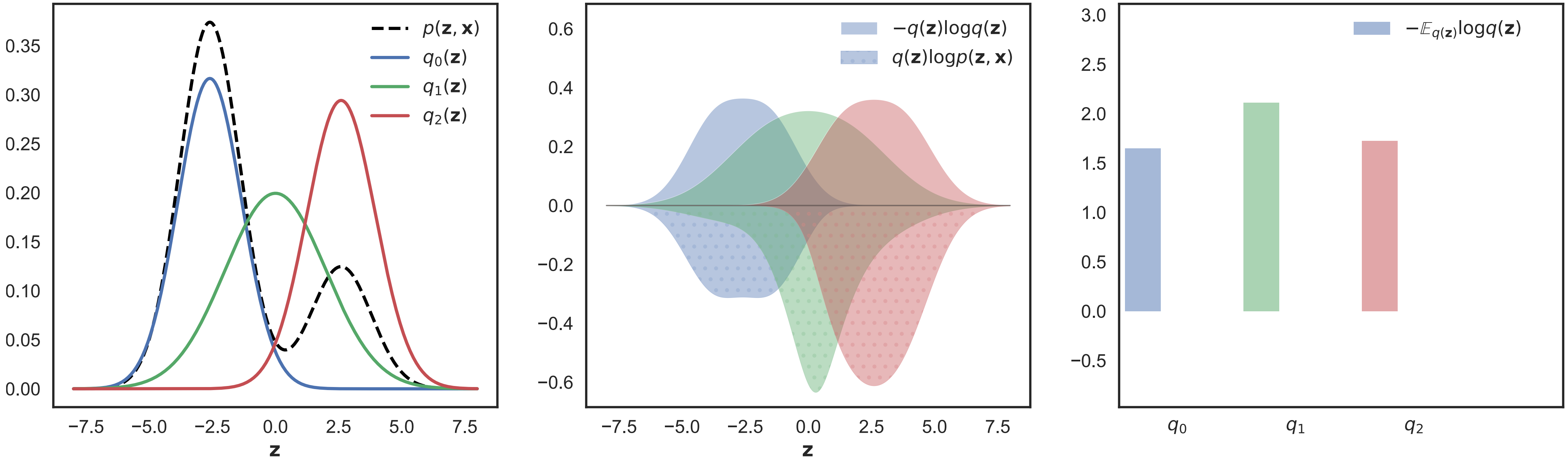


The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)} \left[\log p(z,x) - \log q(z)\right]$$

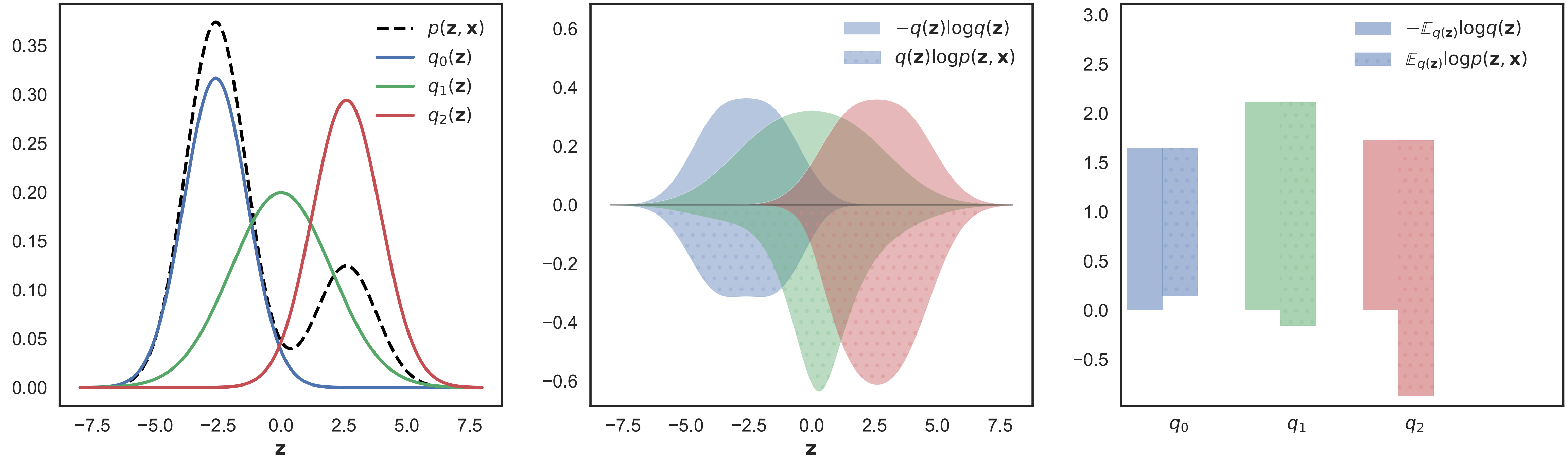


The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)}\left[\log p(z,x) - \log q(z)\right]$$

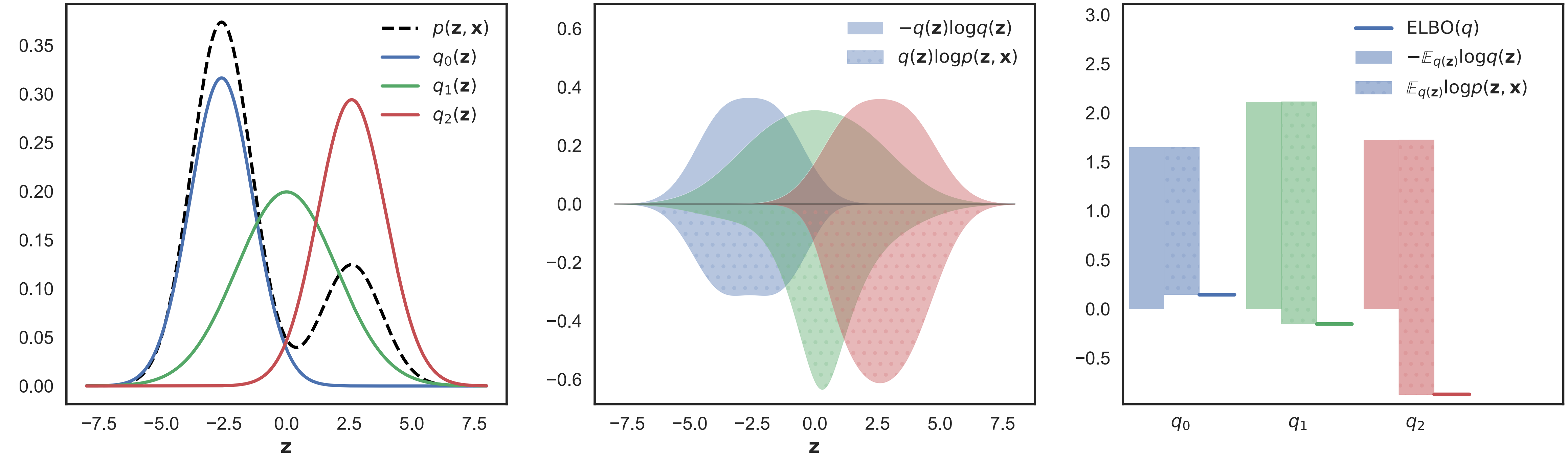


The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)}\left[\log p(z,x) - \log q(z)\right]$$

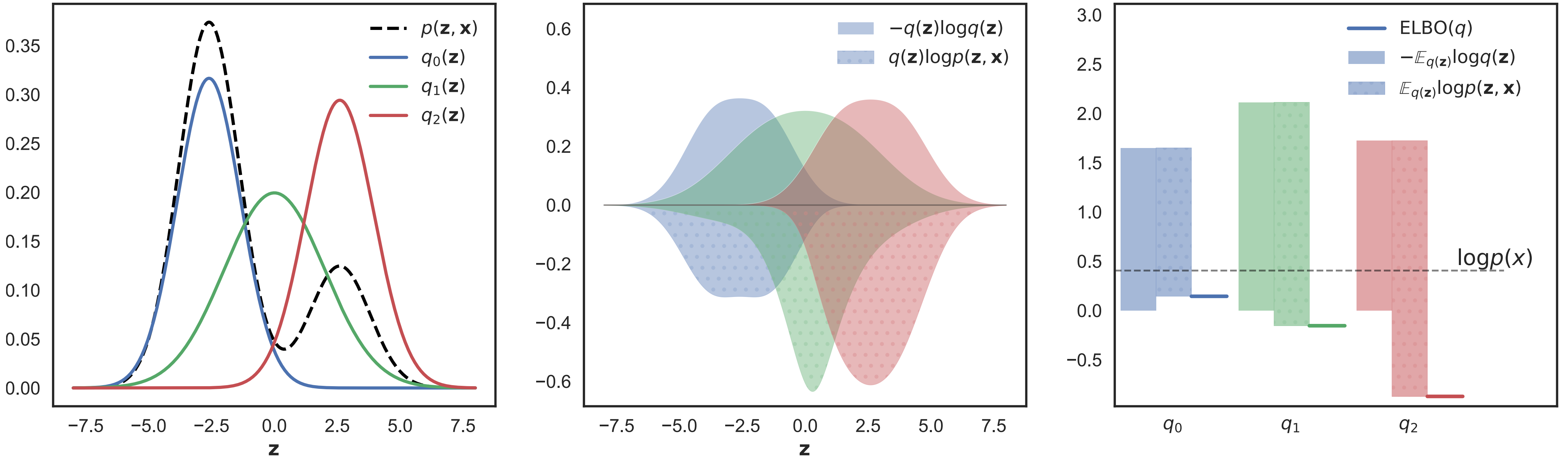


The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)}\left[\log p(z,x) - \log q(z)\right]$$

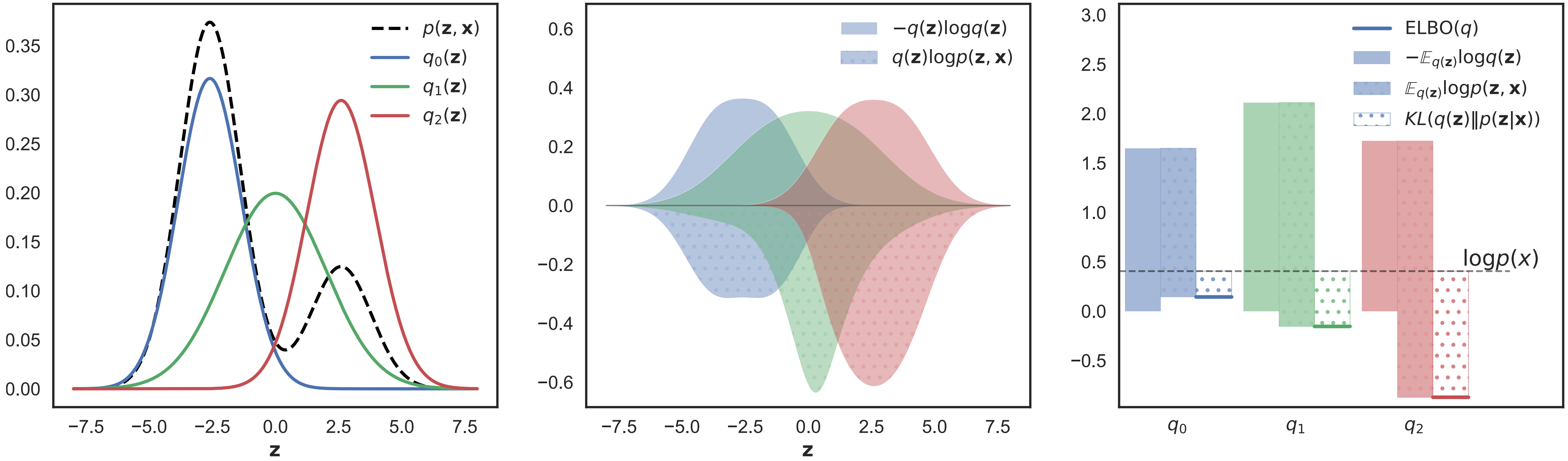


The "ELBO"

is p high?

is q spread?

$$\mathrm{ELBO}(q\|p) = \mathop{\mathbb{E}}_{q(z)}\left[\log p(z,x) - \log q(z)\right]$$



ELBO decomposition

$$\log p(x) = \underbrace{\mathop{\mathbb{E}}_{q(z)} [\log p(z,x) - \log q(z)]}_{\mathrm{ELBO}(q\|p)} + \mathrm{KL}(q(z)\|p(z|x))$$

Lower bound: $\mathrm{ELBO}(q\|p) \leq \log p(x)$

Looseness: $\mathrm{KL}(q(z)\|p(z|x))$

Thought experiment

If $\mathbb{E}R = p(x)$ and $R > 0$,

$$\log p(x) = \underbrace{\mathbb{E}\log R}_{\text{bound}} + \underbrace{\mathbb{E}\log\frac{p(x)}{R}}_{\text{looseness}}$$

If $R = \frac{p(z,x)}{q(z)}, z \sim q$, becomes ELBO decomposition.

But you can use any R....

Variational Autoencoders

$$R = \frac{p_\theta(z, x)}{q(z)}, z \sim q$$

Input $p_\theta(x, z)$ and x.

Find some $q(z)$ and θ to make $\mathbb{E} \log R$ large.

Output θ.

Importance Weighted Autoencoders

$$R_M = \frac{1}{M}\sum_{m=1}^{M}\frac{p_\theta(z_m, x)}{q(z_m)}, z_m \sim q$$

Input $p_\theta(x, z)$ and x.

Find some $q(z)$ and θ to make $\mathbb{E}\log R_M$ large.

Output θ.

(Burda et al., 2015)

Wait a second...

Good old VI: $$\log p(x) = \mathbb{E}\log R + \mathrm{KL}(q(z)\|p(z|x))$$

Learning: $$\mathbb{E}\log R \leq \log p(x)$$

Inference: $$p(z|x) \approx q(z)$$

Now: $$\log p(x) = \mathbb{E}\log R_M + \mathbb{E}\log\frac{p(x)}{R_M}$$

Learning: $$\mathbb{E}\log R_M \leq \log p(x)$$

Inference: $$p(z|x) \approx ???$$

Self-Normalized Importance Sampling

$q_M(z_1):$

$\hat{z}_1, \cdots, \hat{z}_M \sim q(z)$

Sample $m \in \{1, \cdots, M\}$ with $\mathbb{P}(m) \propto p(\hat{z}_m, x)/q(\hat{z}_m)$

Return $z_1 = \hat{z}_m$

$\mathrm{KL}(q_M(z) \| p(z|x)) \longleftarrow$ intractable

Importance weighted VI

$q_M(\cancel{z_1})$:

$\hat{z}_1, \cdots, \hat{z}_M \sim q(z)$

Sample $m \in \{1, \cdots, M\}$ with $\mathbb{P}(m) \propto p(\hat{z}_m, x)/q(\hat{z}_m)$

Return $\cancel{z_1 = \hat{z}_m}$

Importance weighted VI

$q_M(z_{1:M}):$

$\hat{z}_1, \cdots, \hat{z}_M \sim q(z)$

Sample $m \in \{1, \cdots, M\}$ with $\mathbb{P}(m) \propto p(\hat{z}_m, x)/q(\hat{z}_m)$

Return $z_1 = \hat{z}_m$ and $z_{2:M} = \hat{z}_{-m}$

Importance weighted VI

$q_M(z_{1:M})$:

$\hat{z}_1, \cdots, \hat{z}_M \sim q(z)$

Sample $m \in \{1, \cdots, M\}$ with $\mathbb{P}(m) \propto p(\hat{z}_m, x)/q(\hat{z}_m)$

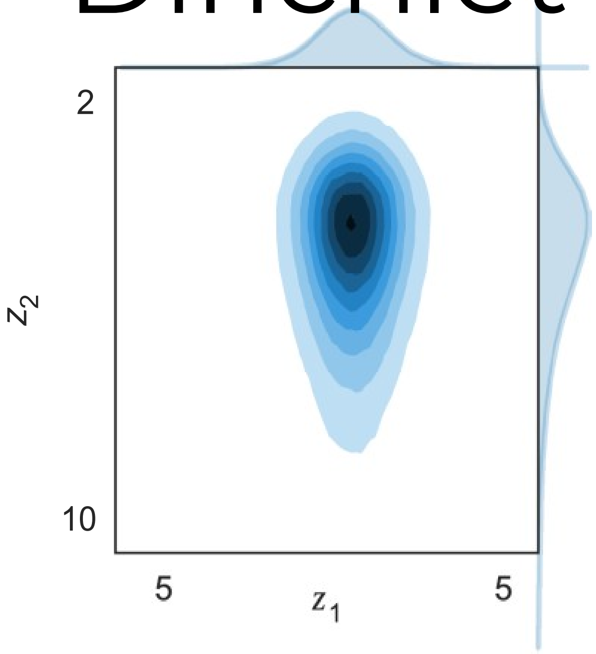
Return $z_1 = \hat{z}_m$ and $z_{2:M} = \hat{z}_{-m}$

$p_M(z_{1:M}, x) := p(z_1, x)q(z_2) \cdots q(z_M)$

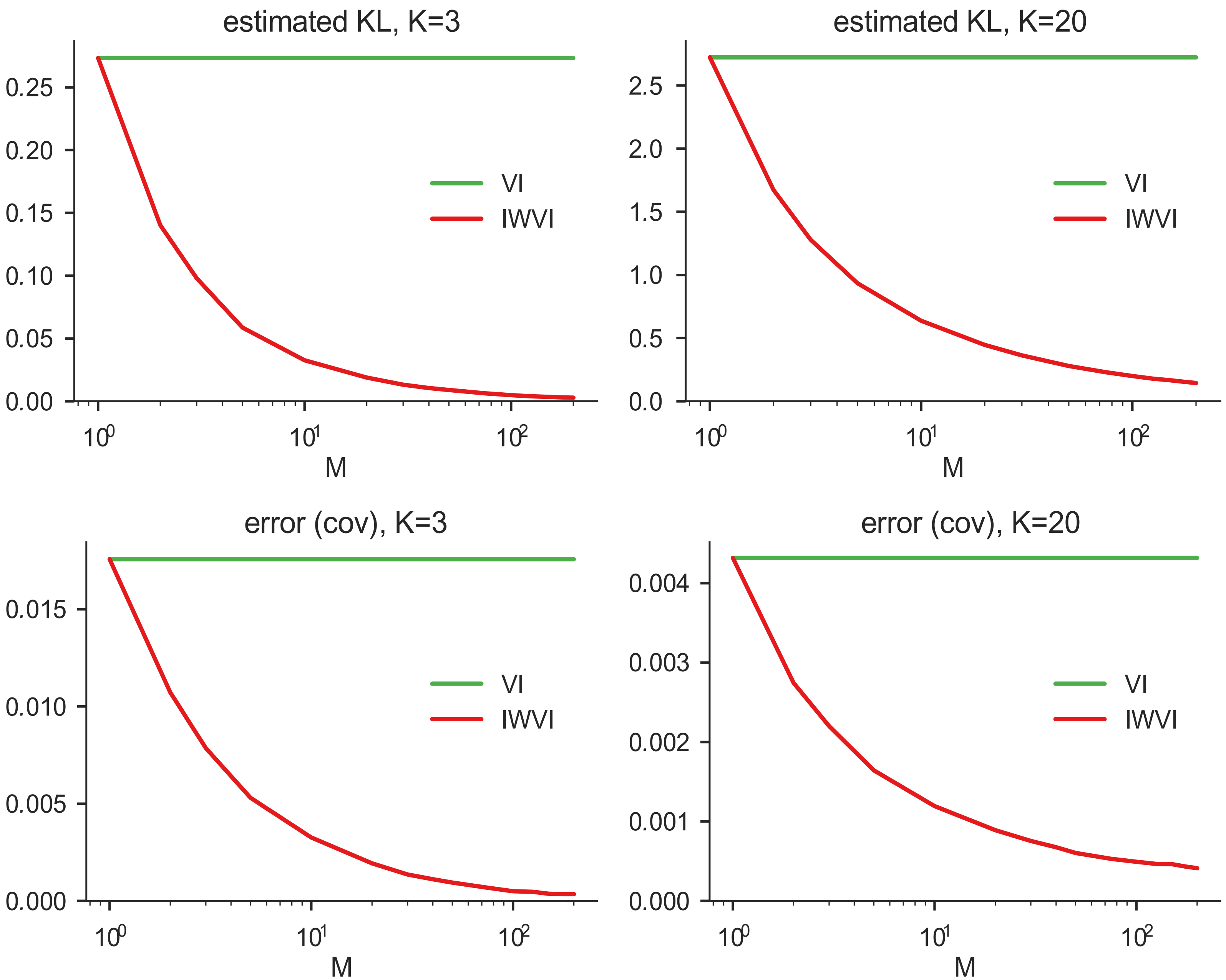
$$\log p(x) = \mathbb{E} \log R_M + \mathrm{KL}(q_M(z_{1:M}) \| p_M(z_{1:M} | x))$$

(Bachman & Precup 2015, Cremer et al., 2017, Naesseth et al. 2018, Le et al. 2018, D. & Sheldon, 2018)

Dirichlet

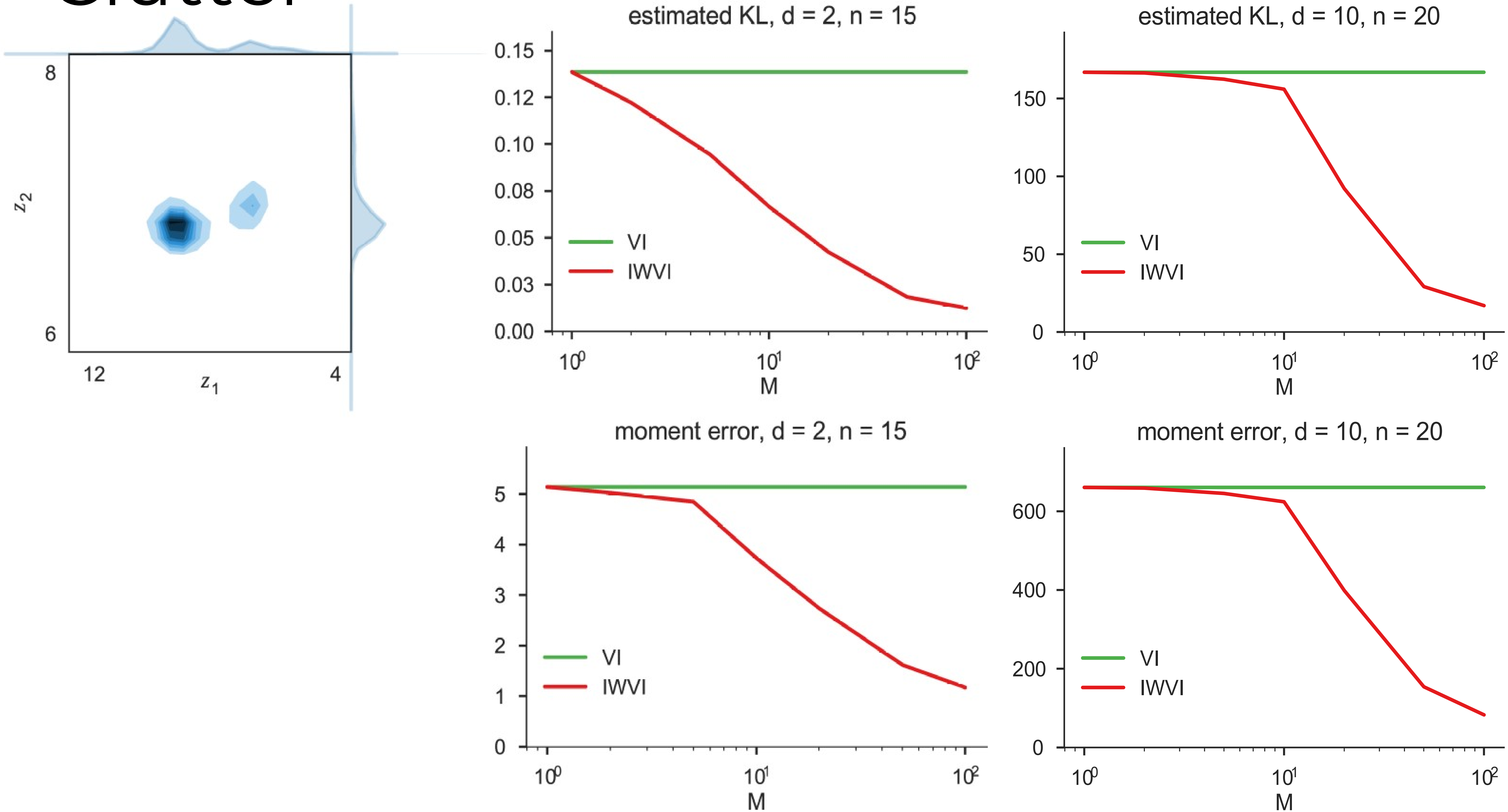


2
10
5
z_1
5
z_2



estimated KL, K=3
0.25
0.20
0.15
0.10
0.05
0.00
10^0
10^1
10^2
M
VI
IWVI
estimated KL, K=20
2.5
2.0
1.5
1.0
0.5
0.0
10^0
10^1
10^2
M
VI
IWVI
error (cov), K=3
0.015
0.010
0.005
0.000
10^0
10^1
10^2
M
VI
IWVI
error (cov), K=20
0.004
0.003
0.002
0.001
0.000
10^0
10^1
10^2
M
VI
IWVI

Clutter



8
6
12
4
z_1
z_2
estimated KL, d = 2, n = 15
0.15
0.12
0.10
0.08
0.05
0.03
0.00
VI
IWVI
10^0
10^1
10^2
M
estimated KL, d = 10, n = 20
150
100
50
0
VI
IWVI
10^0
10^1
10^2
M
moment error, d = 2, n = 15
5
4
3
2
1
0
VI
IWVI
10^0
10^1
10^2
M
moment error, d = 10, n = 20
600
400
200
0
VI
IWVI
10^0
10^1
10^2
M

Huh?

$$\mathbb{E}R = p(x) \implies \log p(x) = \mathbb{E}\log R + \mathbb{E}\log \frac{p(x)}{R}$$

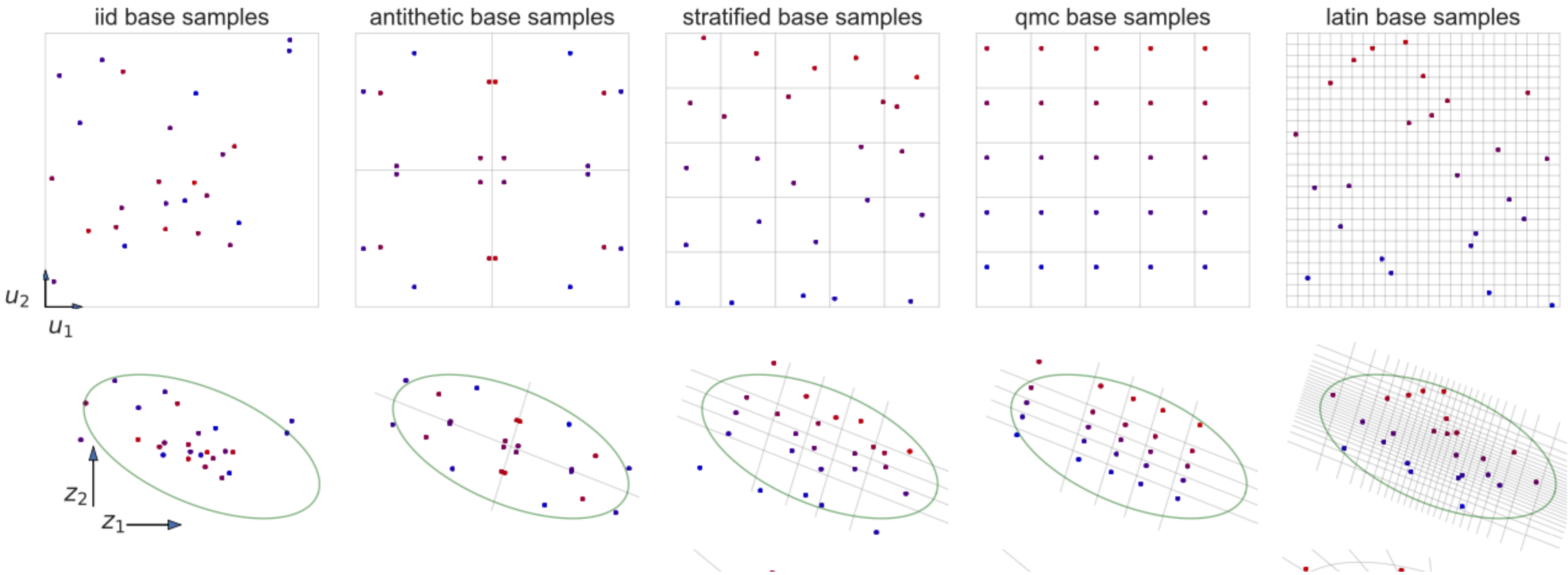
$$R = \frac{p(z,x)}{q(z)} \implies \log p(x) = \mathbb{E}\log R + \mathbb{E}\log \frac{p(z|x)}{q(z)}$$

$$R_M = \frac{1}{M}\sum_{m=1}^{M} \frac{p(z_m,x)}{q(z_m)} \implies \log p(x) = \mathbb{E}\log R_M + \mathbb{E}\log \frac{p_M(z_{1:M}|x)}{q_M(z_{1:M})}$$

augmented distribution?

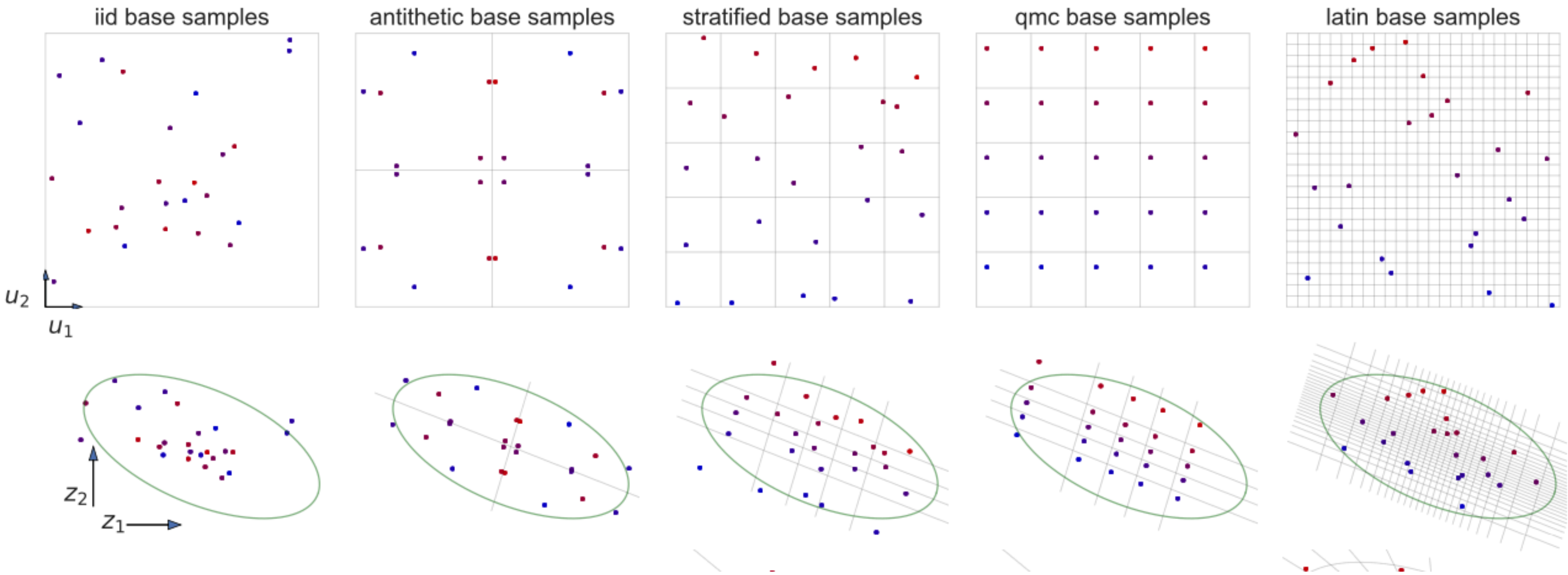
self-normalized importance sampling!?

Other estimators?



$$R_M = \frac{1}{M} \sum_{m=1}^{M} \frac{p(z_m, x)}{q(z_m)}$$

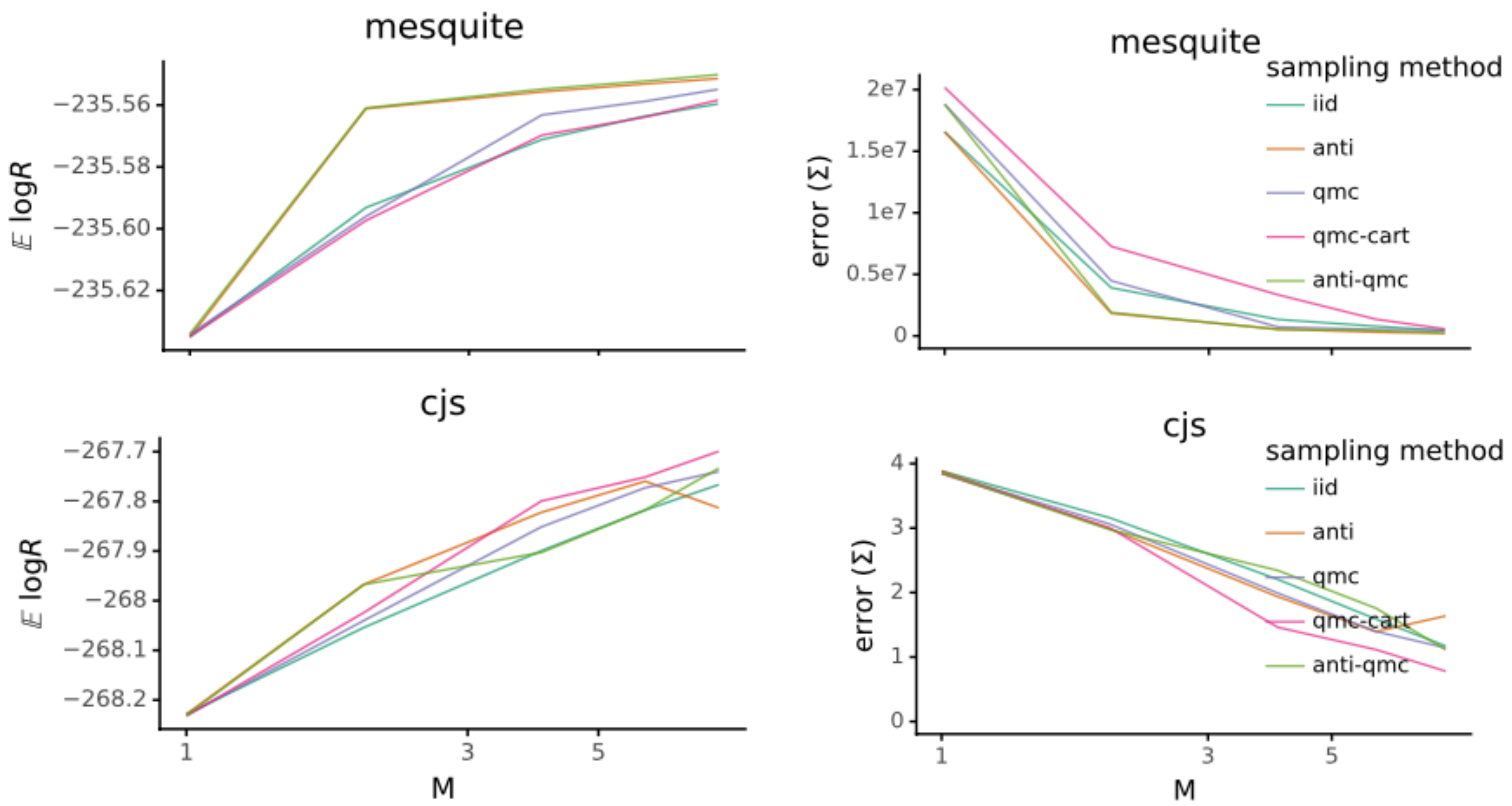
Divide and couple



Can “Automatically” find posterior approximation via “estimator-coupling pairs”

(D. and Sheldon 2019)

Divide and couple

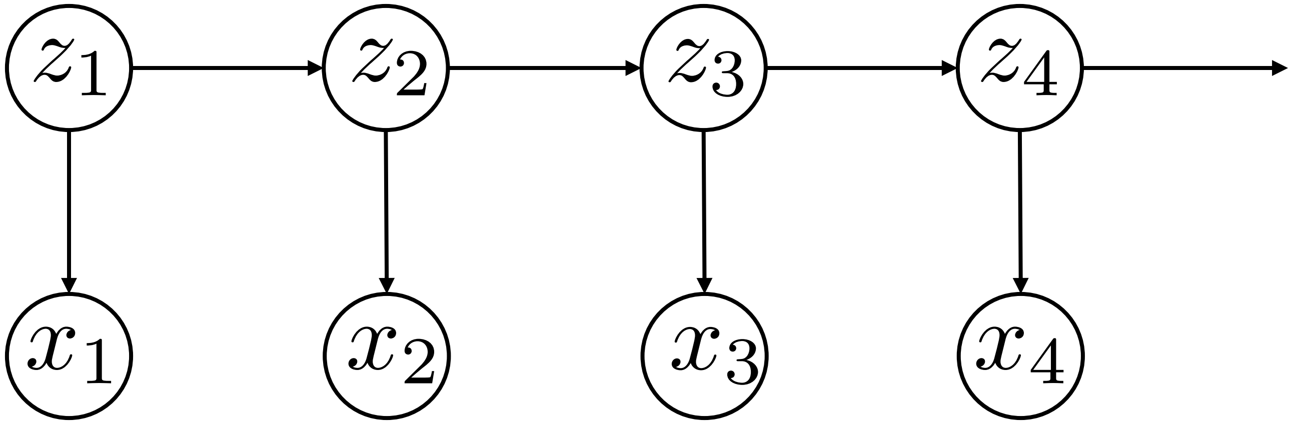


Can “Automatically” find posterior approximation via “estimator-coupling pairs”

(D. and Sheldon 2019)

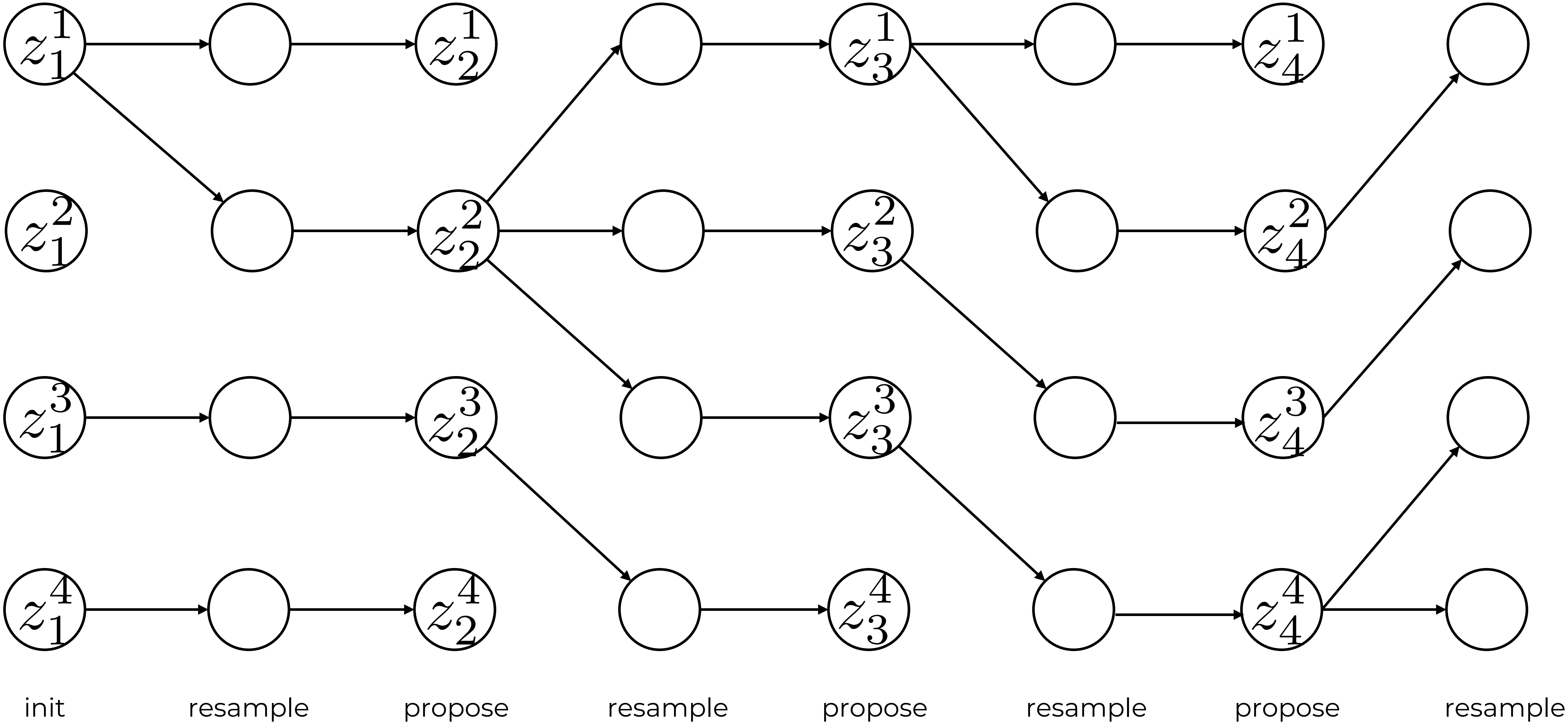
Variational Particle Filtering

State-Space Models



$$p(z_{1:T}, x_{1:T}) = p(z_1, x_1) \prod_{t=2}^{T} p(z_t, x_t | z_{t-1}, x_{t-1})$$

Particle Filter / Sequential Monte Carlo



Particle Filter

$$z_1^m \sim q(z_1), \quad w_1^m = p(z_1^m, x_1)/q(z_1^m)$$

For $t = 2, \cdots, T$:

For $m = 1, \cdots, M$:

Choose parent $n \in \{1, \cdots, M\}$ with $\mathbb{P}(n) \propto w_{t-1}^n$

Sample $z_t^m \sim q(z_t | z_{t-1}^n)$

Set $w_t^m = p(z_t^m, x_t^m | z_{t-1}^n)/q(z_t^m | z_{t-t}^n)$

Variational Particle Filter

$z_1^m \sim q(z_1), \quad w_1^m = p(z_1^m, x_1)/q(z_1^m)$

For $t = 2, \cdots, T:$

For $m = 1, \cdots, M:$

$\boxed{\text{Choose parent } n \in \{1, \cdots, M\} \text{ with } \mathbb{P}(n) \propto w_{t-1}^n}$

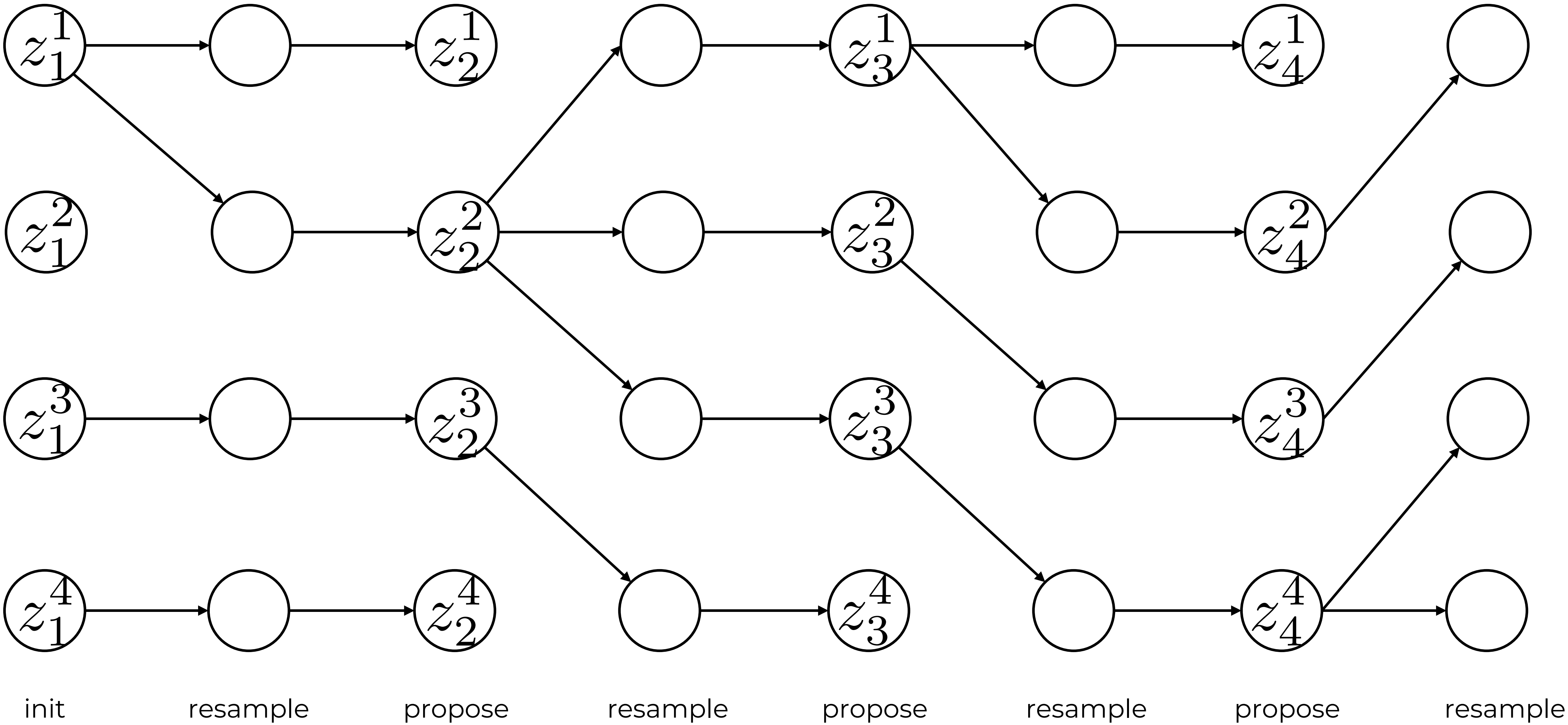
Sample $z_t^m \sim q(z_t | z_{t-1}^n)$

Set $w_t^m = p(z_t^m, x_t^m | z_{t-1}^n)/q(z_t^m | z_{t-t}^n)$

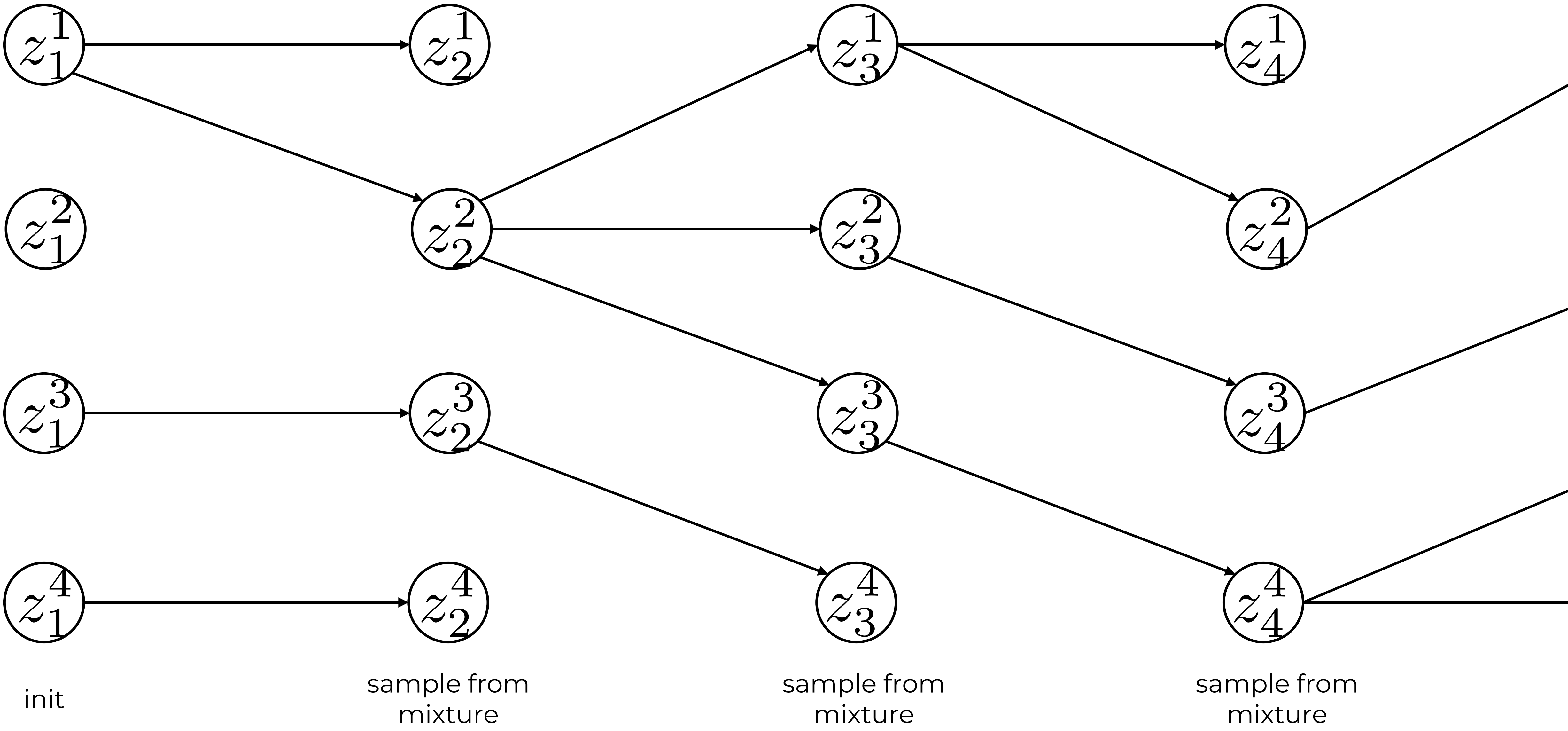
$R_M := \prod_{t=1}^T \frac{1}{M} \sum_{m=1}^M w_t^m \quad \mathbb{E} R_M = p(x)$

Maximize $\mathbb{E} \log R_M$ (Naesseth et al., 2018 Maddison et al., 2017, Le et al., 2018)

Particle Filter



Marginal Particle Filter



Particle Filter

$$z_1^m \sim q(z_1), \quad w_1^m = p(z_1^m, x_1)/q(z_1^m)$$

For $t = 2, \cdots, T$:

For $m = 1, \cdots, M$:

Choose parent $n \in \{1, \cdots, M\}$ with $\mathbb{P}(n) \propto w_{t-1}^n$

Sample $z_t^m \sim q(z_t | z_{t-1}^n)$

Set $w_t^m = p(z_t^m, x_t^m | z_{t-1}^n)/q(z_t^m | z_{t-t}^n)$

Marginal Particle Filter

$$z_1^m \sim q(z_1), \quad w_1^m = p(z_1^m, x_1)/q(z_1^m)$$

For $t = 2, \cdots, T$:

For $m = 1, \cdots, M$:

~~Choose parent~~ $\cancel{n \in \{1, \cdots, M\}}$ ~~with~~ $\cancel{\mathbb{P}(n) \propto w_{t-1}^n}$

~~Sample~~ $\cancel{z_t^m \sim q(z_t | z_{t-1}^n)}$

~~Set~~ $\cancel{w_t^m = p(z_t^m, x_t^m | z_{t-1}^n)/q(z_t^m | z_{t-t}^n)}$

Marginal Particle Filter

$$z_1^m \sim q(z_1), \quad w_1^m = p(z_1^m, x_1)/q(z_1^m)$$

For $t = 2, \cdots, T:$

For $m = 1, \cdots, M:$

Sample z_t^m from *mixture* of $q(z_1|z_{t-1}^1) \cdots q(z_1|z_{t-1}^N)$

with weights $w_{t-1}^1 \cdots w_{t-1}^N$

~~Set~~ $\cancel{w_t^m = p(z_t^m, x_t^m|z_{t-1}^n)/q(z_t^m|z_{t-t}^n)}$

Marginal Particle Filter

$$z_1^m \sim q(z_1), \quad w_1^m = p(z_1^m, x_1)/q(z_1^m)$$

For $t = 2, \cdots, T$:

For $m = 1, \cdots, M$:

Sample z_t^m from *mixture* of $q(z_1|z_{t-1}^1) \cdots q(z_1|z_{t-1}^N)$

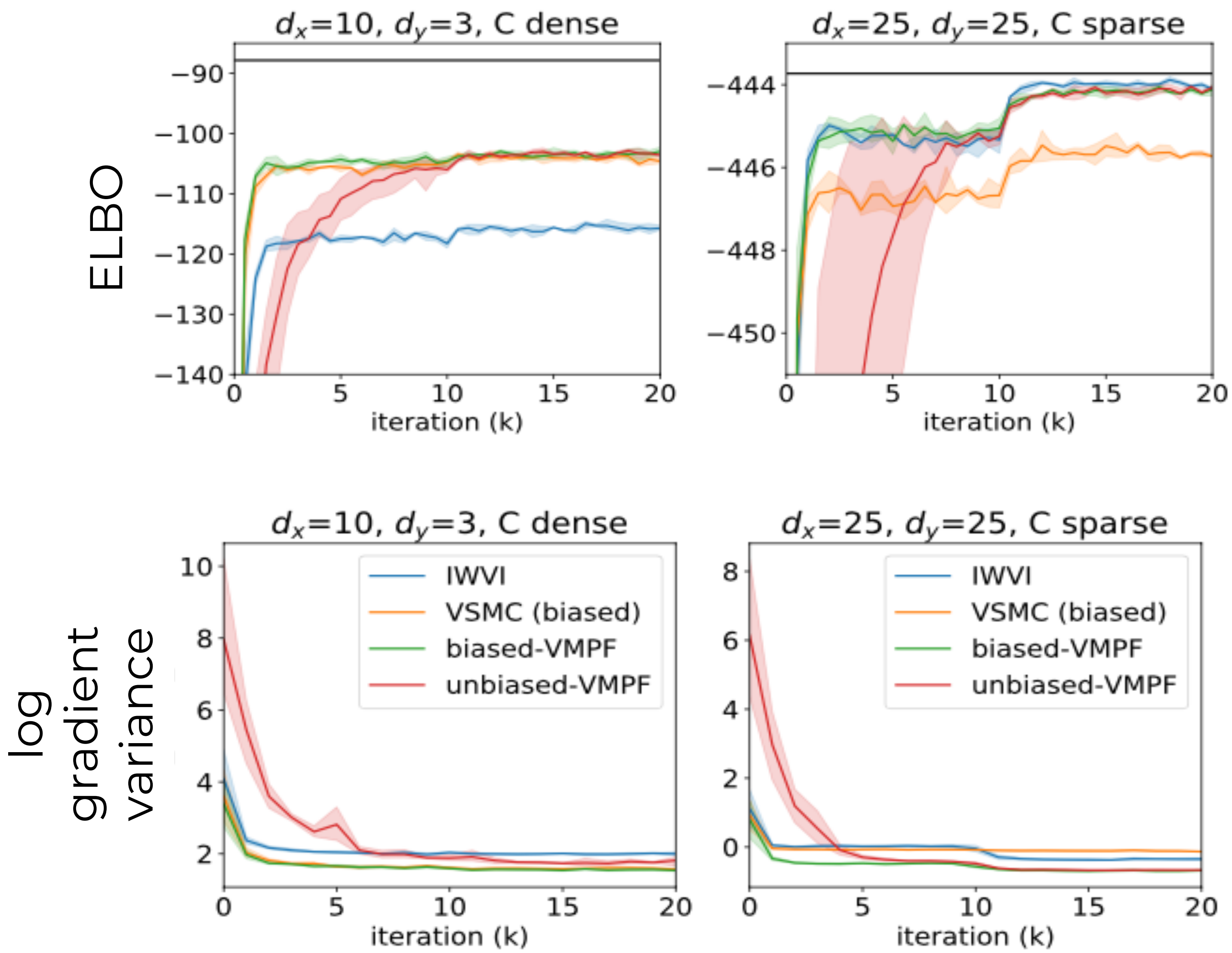
with weights $w_{t-1}^1 \cdots w_{t-1}^N$

Set $w_t^m = \frac{\sum_{n=1}^M w_{t-1}^n p(z_t^m, x_t^m | z_{t-1}^n)}{\sum_{n=1}^M w_{t-1}^n q(z_t^m | z_{t-1}^n)}$

$R_M := \prod_{t=1}^T \frac{1}{M} \sum_{m=1}^M w_t^m \quad \mathbb{E} R_M = p(x)$

Maximize $\mathbb{E} \log R_M$, approximate $p(z_T|x)$ (Lai, D., Sheldon, 2022)

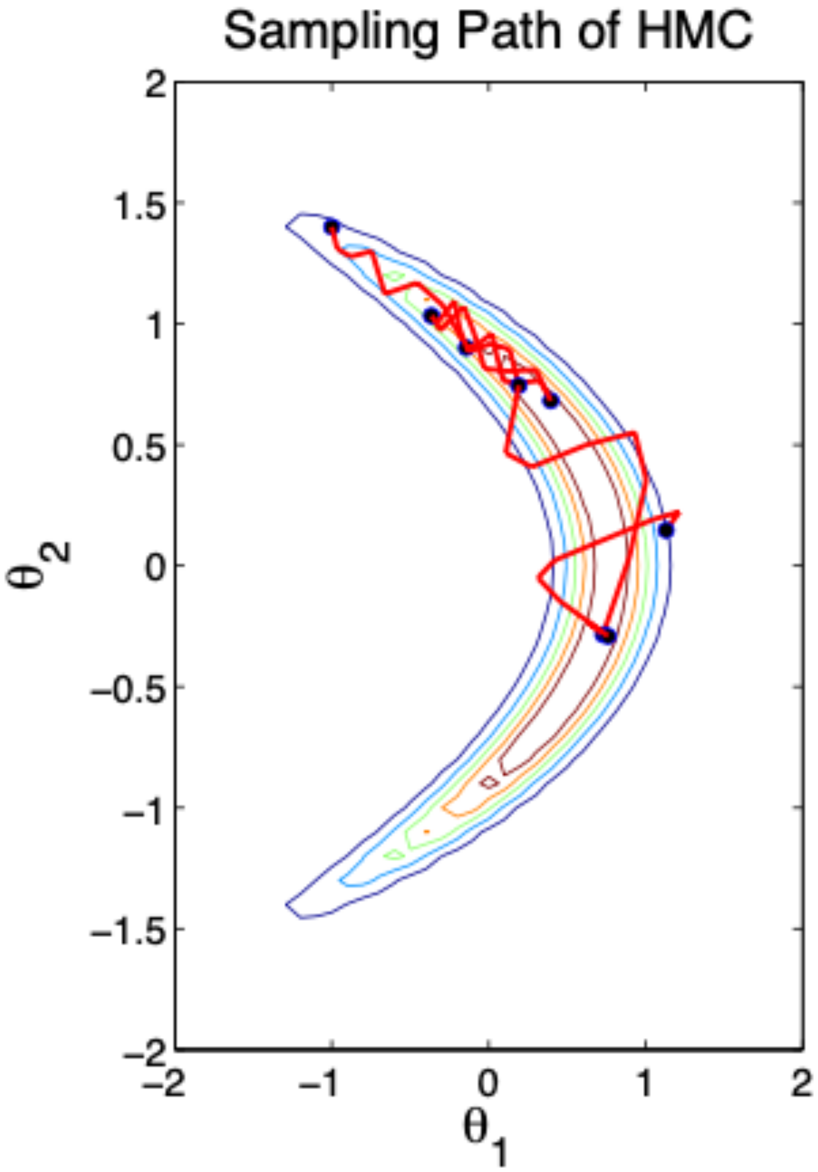
Better… sometimes



IWVI	Importance Weighting
VSMC	(recent method)
VMPF	Variational Marginal Particle Filter (proposed)

Variational Annealing

MCMC



Idea: Find R such that $\mathbb{E}R = p(x)$, optimize $\mathbb{E}\log R_M$

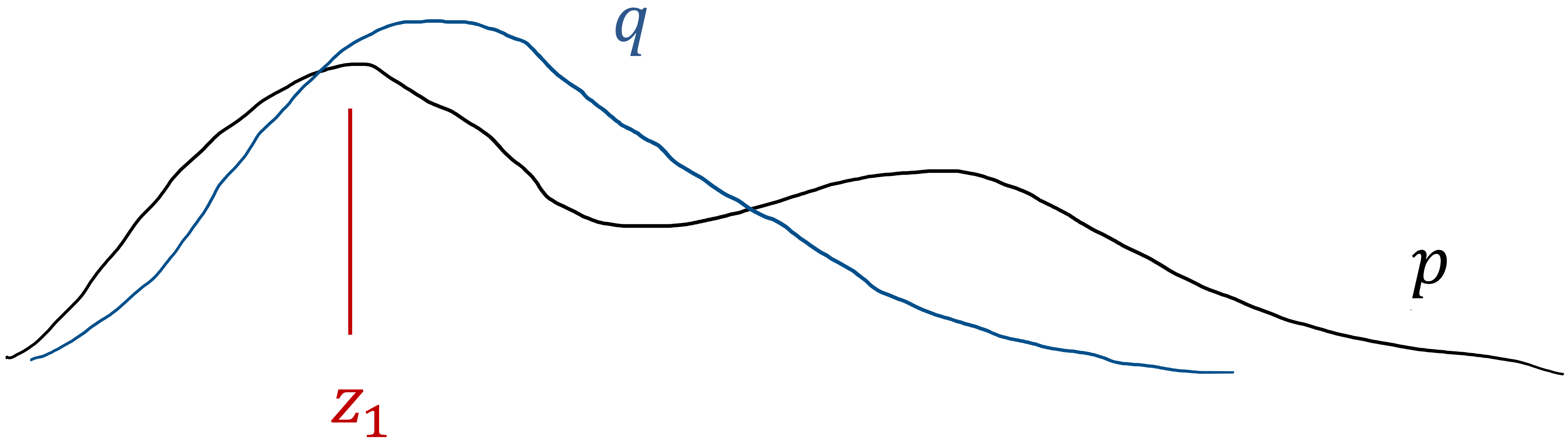
1. Initial dist, mass matrix, step size, annealing schedule

2. Use MCMC to support maximum-likelihood learning

(Figure from Lan et al., 2012)

MCMC

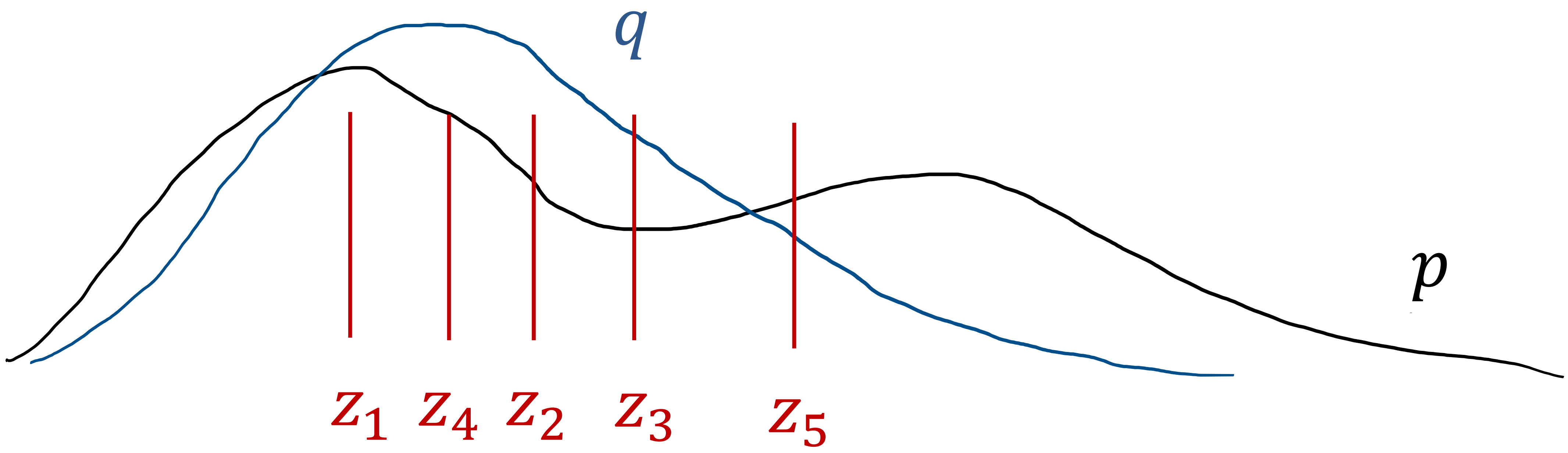
1. Draw $z_1 \sim q(z)$
2. For $k = 1, 2, \cdots K-1$, sample $z_{k+1} \sim T(\cdot \mid z_k)$ where T is one iteration of MCMC



MCMC

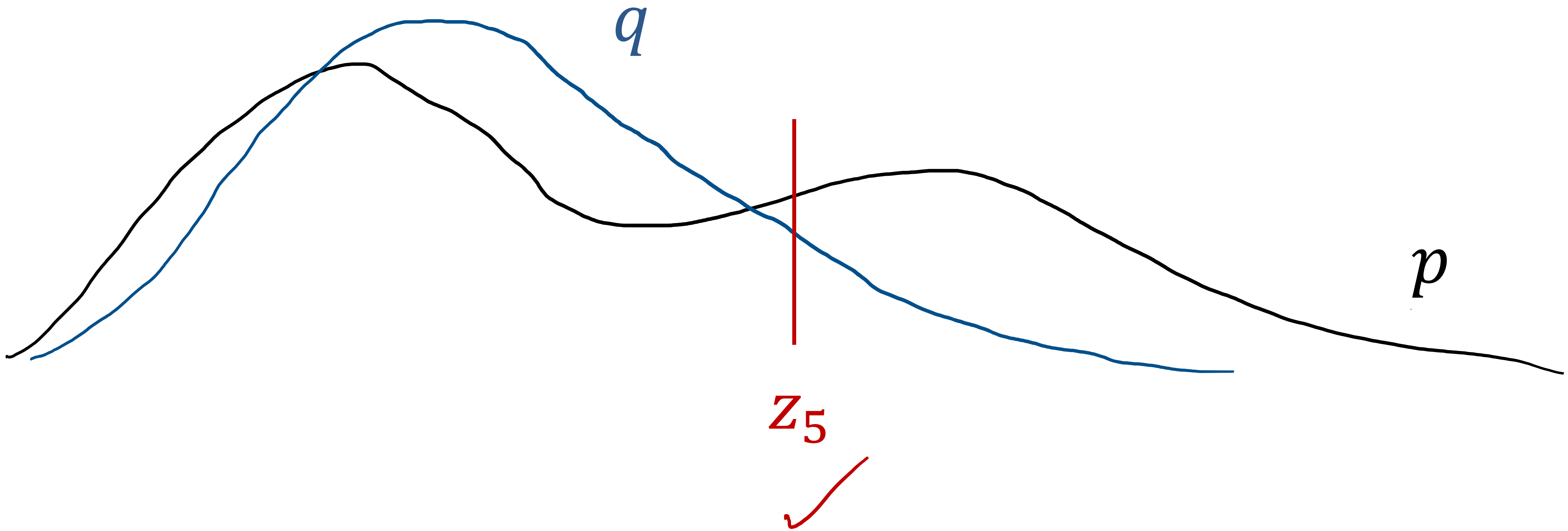
1. Draw $z_1 \sim q(z)$

2. For $k = 1,2, \cdots K-1$, sample $z_{k+1} \sim T(\cdot\,|z_k)$ where T is one iteration of MCMC



MCMC

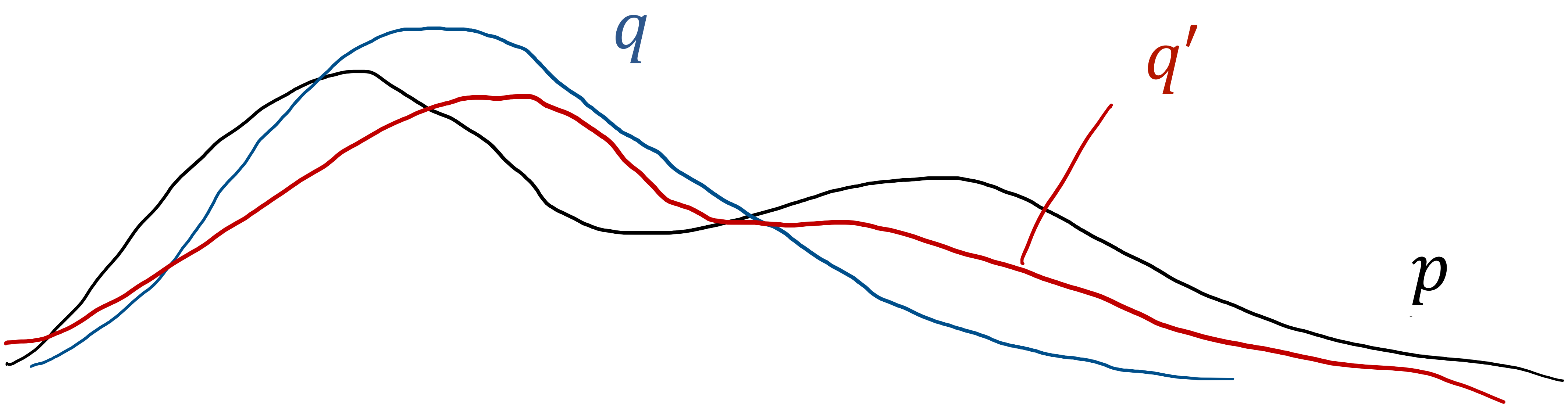
1. Draw $z_1 \sim q(z)$
2. For $k = 1,2,\cdots K-1$, sample $z_{k+1} \sim T(\cdot\,|z_k)$ where T is one iteration of MCMC



MCMC

1. Draw $z_1 \sim q(z)$
2. For $k = 1,2,\cdots K-1$, sample $z_{k+1} \sim T(\cdot\,|z_k)$ where T is one iteration of MCMC

$\big]\ q'$



Augmentation

Problem: Intractable to compute $q'(z_K)$

Instead: Use full trace $q'(z_1, \cdots, z_K)$

- Augment p into $p'(z_1, \cdots, z_K, x)$
- Solve $\min_{q \in \text{Family}} \text{KL}\left(q'(z_1, \cdots, z_K) || p'(z_1, \cdots, z_K | x)\right)$

Bridging the Gap

learned starting dist

one iteration of MCMC

$$q'(z_1, \cdots, z_K) = q\left(z_1\right) \prod_{k=1}^{K-1} q\left(z_{k+1} | z_k\right)$$

$$p'(z_1, \cdots, z_K, x) = p\left(z_K, x\right) \prod_{k=1}^{K-1} p\left(z_k | z_{k+1}\right)$$

original target

learned inverse dynamics

Good: Can optimize “end to end” (dropping accept/reject)
Bad: Learning inverse dynamics is hard.

(Salimans et al. 2015)

Annealed Importance Sampling

learned starting dist

one iteration of MCMC on π_m

$$q'(z_1, \cdots, z_K) = q\left(z_1\right) \prod_{k=1}^{K-1} q\left(z_{k+1}|z_k\right)$$

$$p'(z_1, \cdots, z_K, x) = p\left(z_K, x\right) \prod_{k=1}^{K-1} p\left(z_k|z_{k+1}\right)$$

original target

"reversal" of $q\left(z_{m+1}|z_m\right)$ w.r.t. π_m

Bridging densities $\pi_1 \cdots \pi_K$ go from $q(z_1)$ to $p(z_K|x)$.

Good: $p'(z_1, \cdots, z_K, x)/q'(z_1, \cdots, z_K)$ has simple closed form.

Good: No need to learn inverse dynamics.

Bad: Must use accept/reject steps. Hard to optimize.

(Jarzynski 1997, Neal 2001)

Uncorrected Hamiltonian Annealing

$$q'(z_1,\cdots,z_K) = \underbrace{q\left(z_1\right)}_{\text{learned starting dist}} \prod_{k=1}^{K-1} \underbrace{q\left(z_{k+1}|z_k\right)}_{\text{one iteration of HMC on } \pi_m \text{ with NO CORRECTION}}$$

$$p'(z_1,\cdots,z_K,x) = \underbrace{p\left(z_K,x\right)}_{\text{original target}} \prod_{k=1}^{K-1} \underbrace{p\left(z_k|z_{k+1}\right)}_{\text{“run HMC backwards” for one iteration}}$$

Bridging densities $\pi_1 \cdots \pi_K$ go from $q(z_1)$ to $p(z_K|x)$.

Good: $p'(z_1,\cdots,z_K,x)/q'(z_1,\cdots,z_K)$ has simple closed form.

Good: No need to learn inverse dynamics.

Good: Fully differentiable, can optimize “end to end”.

(Geffner & D. 2021, Zhang et al. 2021)

Algorithm

Sample state $z_1 \sim q$ and momentum $\rho_1 \sim S$.

Initialize ELBO estimator as $\mathcal{L} \leftarrow -\log q(z_1)$.

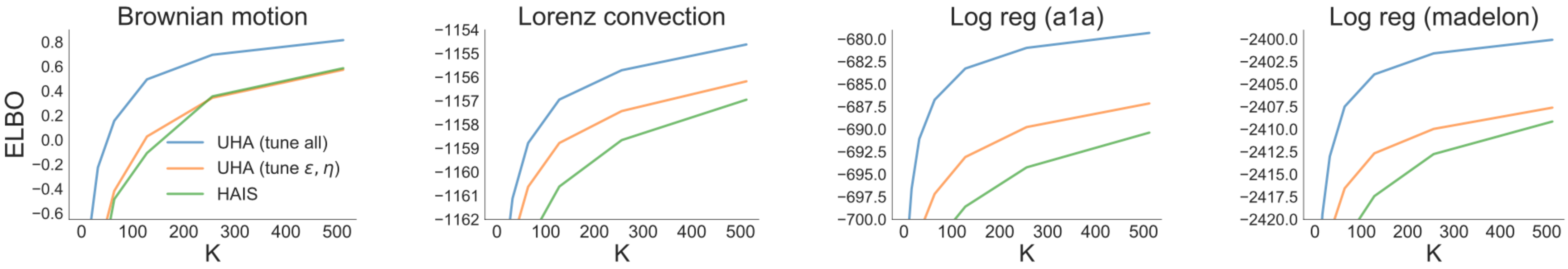
For $k = 1,2,\cdots,K-1$:

- Sample new momentum ρ'.
- $(z_{k+1}, \rho_{k+1}) \leftarrow$ HMC with target π_k and starting point (z_k, ρ').
- Update estimator as $\mathcal{L} \leftarrow \mathcal{L} + \log S(\rho_{k+1}) - \log S(\rho')$

Update estimator as $\mathcal{L} \leftarrow \mathcal{L} + \log p(z_K, x)$

Can compute gradients for initial distribution q, bridging schedule, HMC hyperparameters, etc.

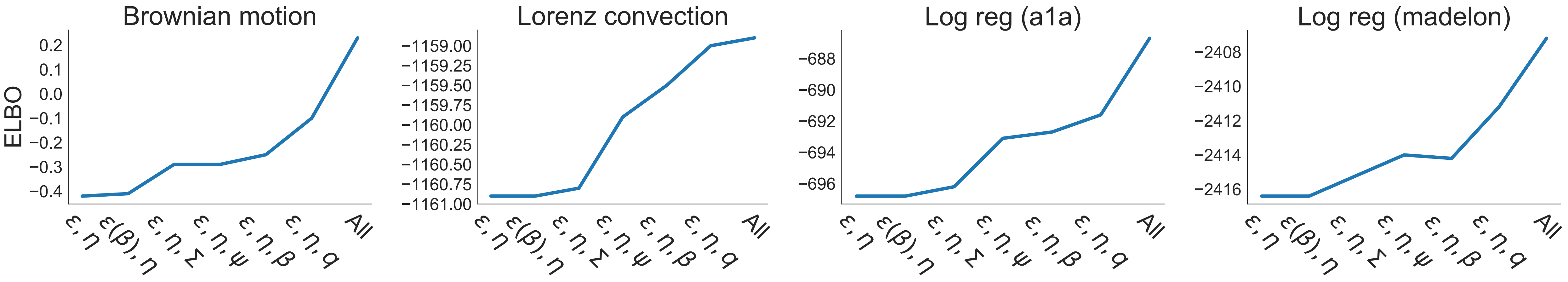
Tuning stuff is good



ε – step size of HMC dynamics
η – damping coefficient
Σ – moment covariance
β – temperature schedule
ψ – “full rank” temperature schedule
q – initial distribution

(K = total # of evaluations of $p(z, x)$ per ELBO est.)

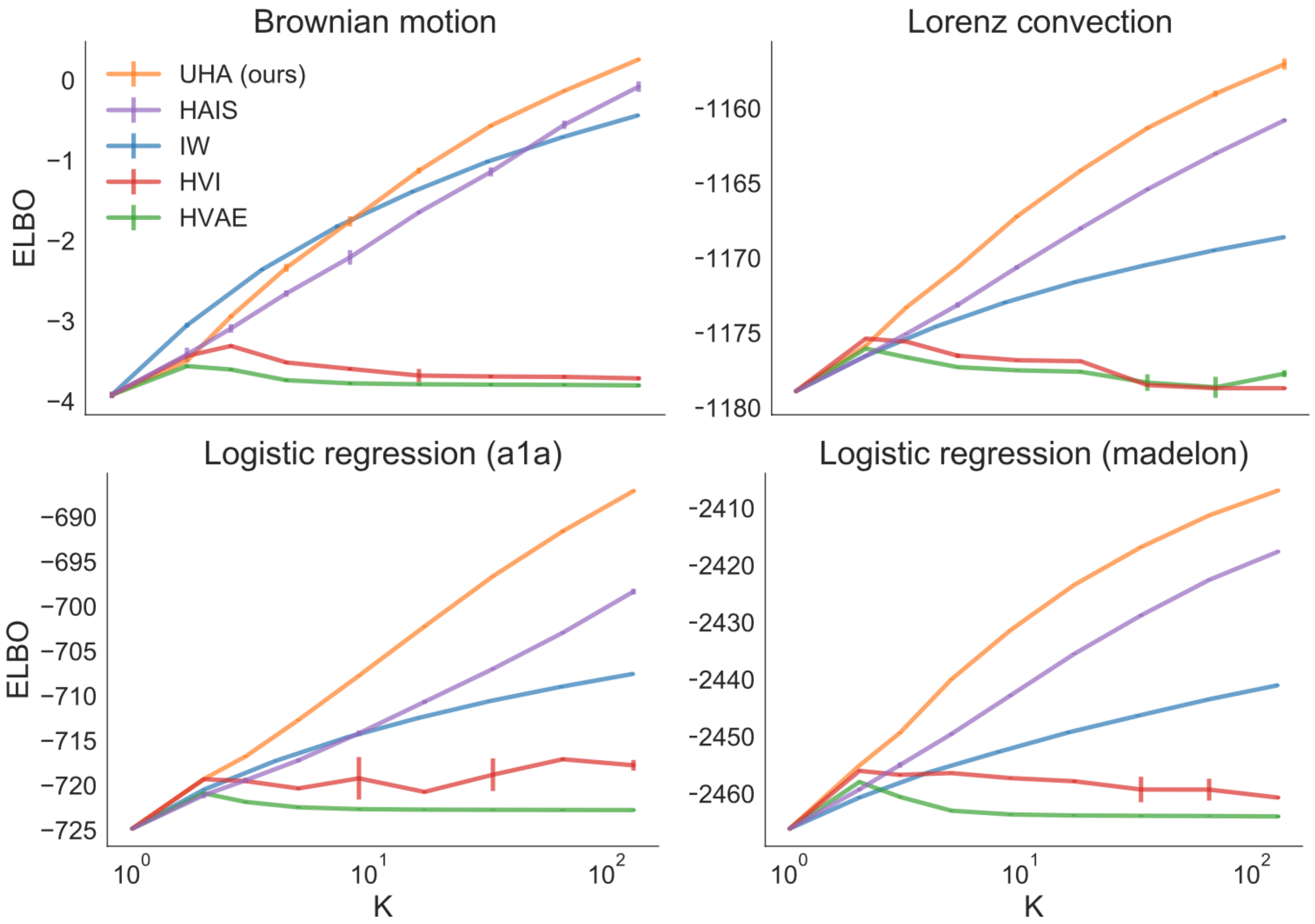
Tuning more stuff is good



(K = 64)

ε – step size of HMC dynamics
η – damping coefficient
Σ – moment covariance
β – temperature schedule
ψ – “full rank” temperature schedule
q – initial distribution

Compares well to baselines



UHA	Our algorithm
HAIS	Annealed Importance Sampling using HMC dynamics
IW	Importance Weighting
HVI	“Bridging the gap” using HMC dynamics
HVAE	(Recent algorithm)

VAE training

ELBO on test set

		$K=1$	$K=8$	$K=16$	$K=32$	$K=64$
mnist	UHA	-93.4	-89.8	-88.8	-88.1	-87.6
	IW	-93.4	-90.5	-89.9	-89.4	-89.0
letters	UHA	-137.9	-133.5	-132.3	-131.5	-130.9
	IW	-137.9	-134.6	-133.9	-133.2	-132.7
kmnist	UHA	-184.2	-176.6	-174.6	-173.2	-171.6
	IW	-184.2	-179.7	-178.7	-177.8	-177.0

log-likelihood on test set

		$K=1$	$K=8$	$K=16$	$K=32$	$K=64$
mnist	UHA	-88.5	-87.5	-87.2	-87.0	-86.9
	IW	-88.5	-87.6	-87.5	-87.3	-87.2
letters	UHA	-131.9	-130.7	-130.3	-130.1	-129.9
	IW	-131.9	-130.9	-130.7	-130.6	-130.4
kmnist	UHA	-174.3	-172.2	-171.6	-171.2	-170.2
	IW	-174.3	-173.0	-172.6	-172.4	-172.2

Thank you!

Dan Sheldon

Jinlin Lai

Tomas Geffner

Publications

Importance Weighted Variational Inference, NeurIPS, 2018

Divide and Couple: Using Monte Carlo Variational Objectives for Posterior Approximation, NeurIPS 2018

Variational Marginal Particle Filters, AISTATS 2022

MCMC Variational Inference via Uncorrected Hamiltonian Annealing, NeurIPS 2021