Sequential Monte Carlo For Amortized Variational Inference

Declan McNamara

Department of Statistics
University of Michigan

August 9, 2023

Collaborators

Jeffrey Regier

Jackson Loper

Outline

1. Background
   - Variational Inference
   - Amortized Variational Inference
2. SMC-Wake
   - Forward KL Objective
   - Experiments

Background

Bayesian Inference

- Given a generative model $p(\theta, x)$, the posterior distribution (density) can be computed by Bayes' rule via

$$p(\theta \mid x) = \frac{p(\theta, x)}{p(x)}.$$

Bayesian Inference

- Given a generative model $p(\theta, x)$, the posterior distribution (density) can be computed by Bayes' rule via

$$p(\theta \mid x) = \frac{p(\theta, x)}{p(x)}.$$

- The *evidence* $p(x) = \int p(\theta, x) d\theta$ is usually computationally intractable, so we resort to approximate Bayesian methods.

Variational Inference

- Variational inference (VI) uses optimization to select a parametric distribution $q_{\mathrm{VI}} \in \mathcal{Q}$ to approximate the posterior distribution $p(\theta \mid x)$.

Variational Inference

- Variational inference (VI) uses optimization to select a parametric distribution $q_{\mathrm{VI}} \in \mathcal{Q}$ to approximate the posterior distribution $p(\theta \mid x)$.
- The choice of the objective function L is an area of active research.

Variational Inference

- Variational inference (VI) uses optimization to select a parametric distribution $q_{\mathrm{VI}} \in \mathcal{Q}$ to approximate the posterior distribution $p(\theta \mid x)$.
- The choice of the objective function L is an area of active research.
- What if we want to perform VI for $x_1, \ldots, x_n$, for n large?

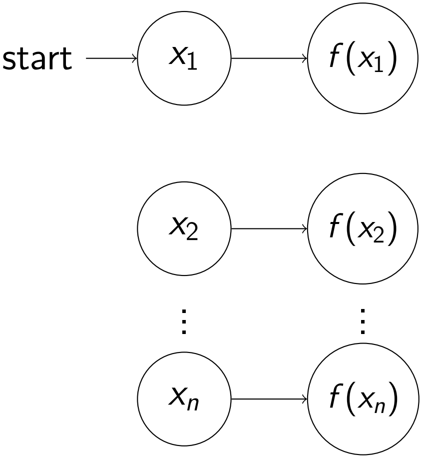
Amortized Inference

- Amortized VI fits an *inference network* f that maps observations $x \in \mathcal{X}$ to variational distributions $q \in \mathcal{Q}$.

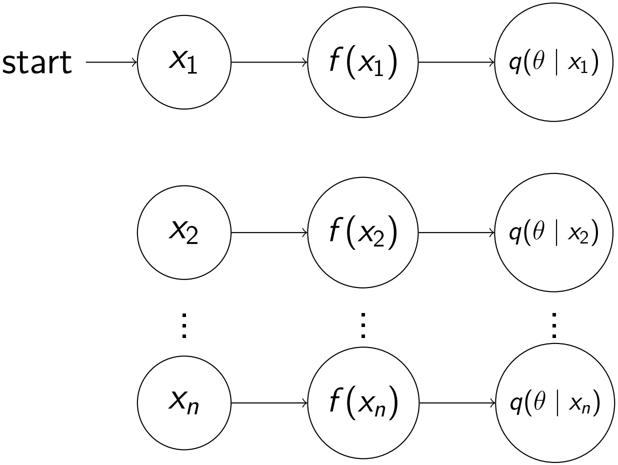
Amortized Inference

- Amortized VI fits an *inference network* f that maps observations $x \in \mathcal{X}$ to variational distributions $q \in \mathcal{Q}$.
- Ideally, $f(x)$ defines a distribution "close" to the posterior $p(\theta \mid x)$ for each $x \in \mathcal{X}$.

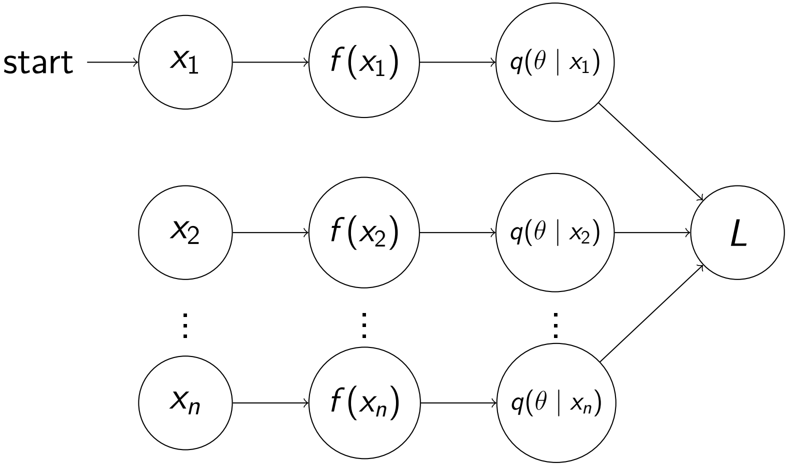
Amortized Inference



Amortized Inference



Amortized Inference



Example

- Suppose that the variational family $\mathcal{Q}$ is taken to be a bivariate Gaussian distribution on 2-dimensional θ.

Example

- Suppose that the variational family $\mathcal{Q}$ is taken to be a bivariate Gaussian distribution on 2-dimensional θ.
- For each observation x, a neural network f would have 5 scalar outputs that define a distribution on θ,

$$f(x) = (\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) \mapsto \mathcal{N}\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \right)$$

Objectives

We let $q_\phi(\theta \mid x)$ denote an amortized variational posterior, corresponding to a neural network with parameters ϕ.

Objectives

$$q^{\mathrm{VI}}(\theta) = \underset{q \in \mathcal{Q}}{\operatorname{argmin}} \; L(q(\theta), \pi(\theta), p(x \mid \theta))$$

Objectives

$$q^{\mathrm{VI}}(\theta) = \operatorname*{argmin}_{q \in \mathcal{Q}} \; L(q(\theta), \pi(\theta), p(x \mid \theta))$$

$$\downarrow$$

$$q_{\phi}^{\mathrm{AVI}}(\theta) = \operatorname*{argmin}_{\phi \in \Phi} \; \frac{1}{n} \sum_{i=1}^{n} L(q_{\phi}(\theta \mid x_i), \pi(\theta), p(x_i \mid \theta))$$

SMC-Wake

The Forward KL Divergence

- We target minimization of the forward KL divergence

$$L(q, \pi, p) = \mathrm{KL}\left[p(\theta \mid x) || q(\theta \mid x)\right].$$

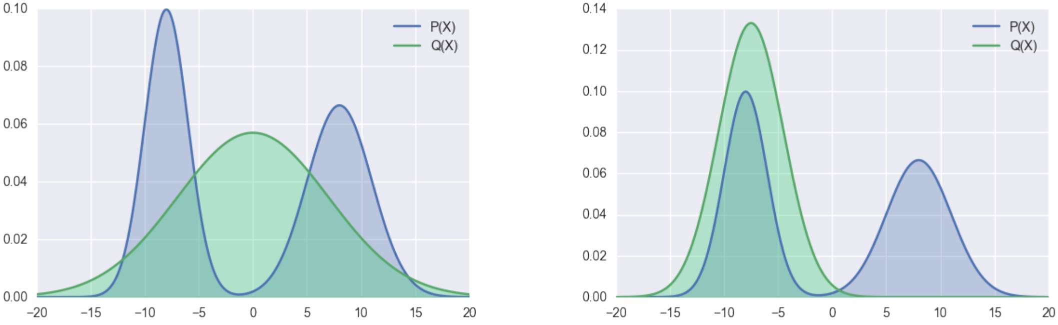
The Forward KL Divergence

- We target minimization of the forward KL divergence

$$L(q, \pi, p) = \mathrm{KL}\left[p(\theta \mid x) || q(\theta \mid x)\right].$$

- VI often uses the evidence lower bound (ELBO) to target the reverse KL divergence.

Forward vs. Reverse KL



(a) Forward KL (b) Reverse KL

Image source: `https://agustinus.kristia.de/techblog/2016/12/21/forward-reverse-kl/`

Particle Approximations

- Estimating the forward KL objective

$$\mathbb{E}_{p(\theta|x)} \log \frac{p(\theta \mid x)}{q_\phi(\theta \mid x)}$$

or its gradient

$$-\mathbb{E}_{p(\theta|x)} \nabla_\phi \log q_\phi(\theta \mid x)$$

requires sampling from the exact posterior.

Particle Approximations

- Estimating the forward KL objective

$$\mathbb{E}_{p(\theta|x)} \log \frac{p(\theta \mid x)}{q_\phi(\theta \mid x)}$$

or its gradient

$$-\mathbb{E}_{p(\theta|x)} \nabla_\phi \log q_\phi(\theta \mid x)$$

requires sampling from the exact posterior.

- The "Wake" algorithm uses self-normalized importance sampling (IS) to estimate the objective.

Particle Approximations

- Estimating the forward KL objective

$$\mathbb{E}_{p(\theta|x)} \log \frac{p(\theta \mid x)}{q_\phi(\theta \mid x)}$$

or its gradient

$$-\mathbb{E}_{p(\theta|x)} \nabla_\phi \log q_\phi(\theta \mid x)$$

requires sampling from the exact posterior.

- We propose using tools from the Sequential Monte Carlo (SMC) literature to estimate this objective.

SMC

- SMC uses a series of intermediate distributions to transition a weighted particle set from a base distribution to a target distribution.

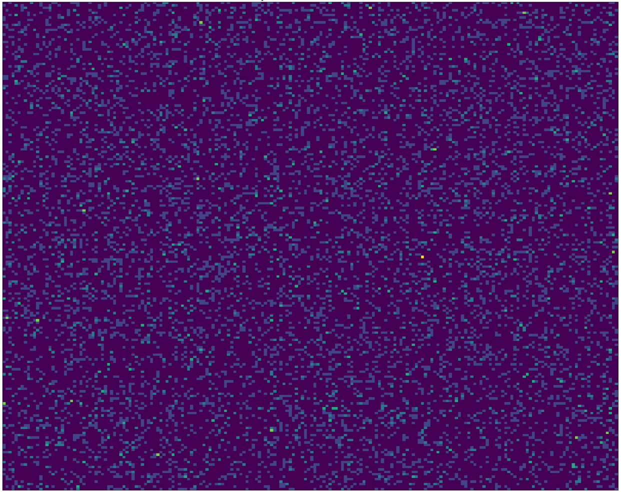
SMC

- SMC uses a series of intermediate distributions to transition a weighted particle set from a base distribution to a target distribution.
- Weighted samples from the base distribution are resampled, extended, and reweighted.

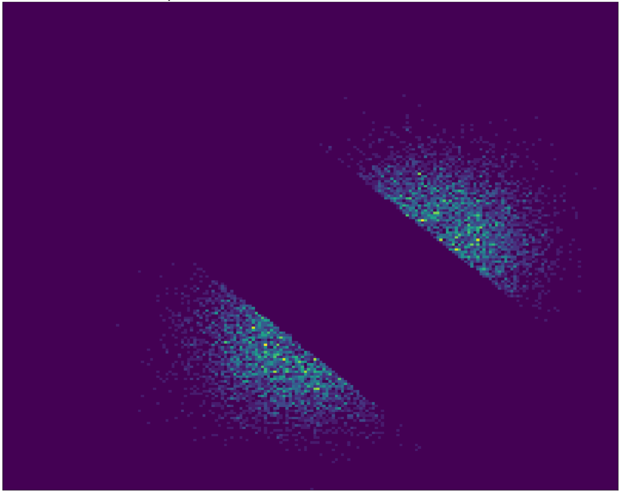
SMC

- SMC uses a series of intermediate distributions to transition a weighted particle set from a base distribution to a target distribution.
- Weighted samples from the base distribution are resampled, extended, and reweighted.
- While SMC is commonly used for state-space models, it can be used to target *any* distribution.

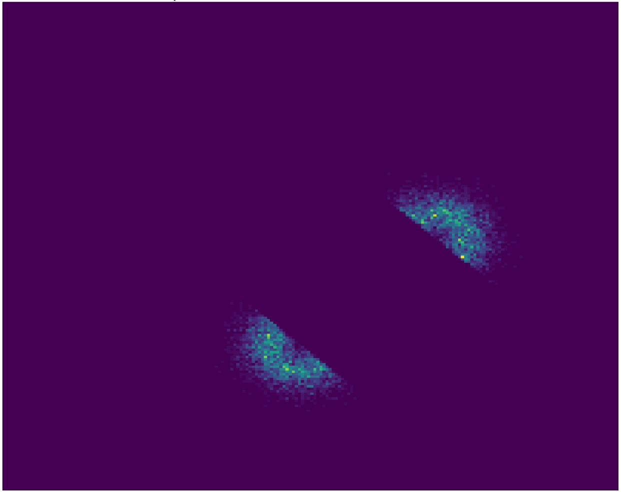
Temperature $\tau = 0.0$



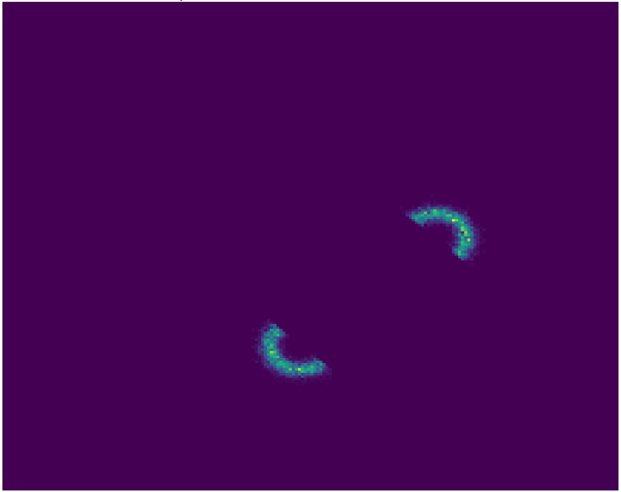
Temperature $\tau = 0.005289797269164934$



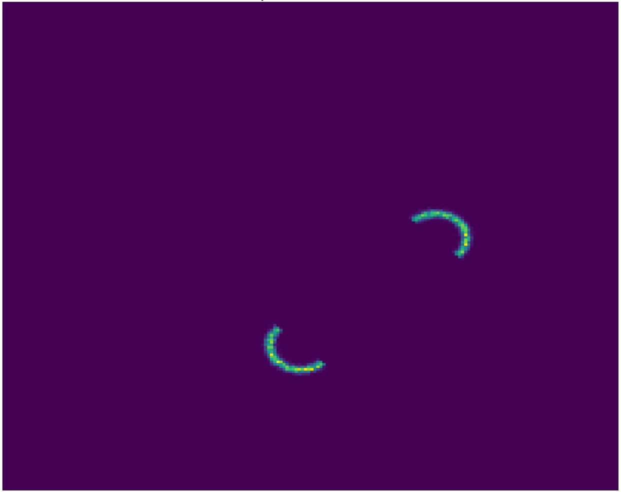
Temperature $\tau = 0.028516511264105\,06$



Temperature $\tau = 0.2099392022875901$



Temperature $\tau = 1.0$



Likelihood-Tempered SMC

- *Auxiliary* variables $\theta_1, \ldots, \theta_{T-1}$ are introduced, with θ_T the target for inference.

Likelihood-Tempered SMC

- *Auxiliary* variables $\theta_1, \ldots, \theta_{T-1}$ are introduced, with θ_T the target for inference.
- Likelihood-tempered SMC imposes the joint distribution

$$p(\theta_k, x) \propto \pi(\theta_k) p(x \mid \theta_k)^{\tau_k}.$$

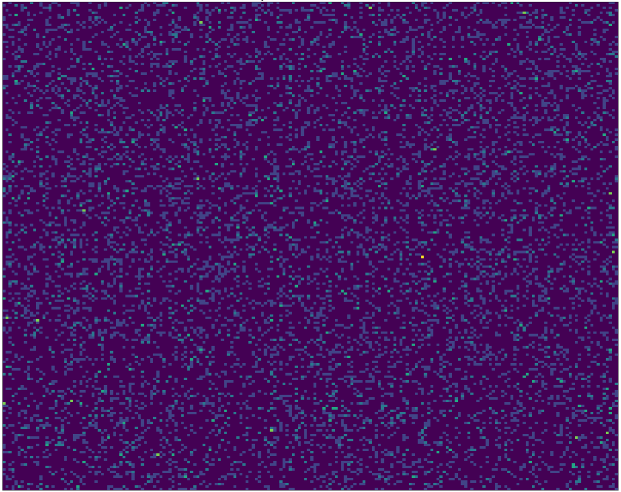
Likelihood-Tempered SMC

- *Auxiliary* variables $\theta_1, \ldots, \theta_{T-1}$ are introduced, with θ_T the target for inference.
- Likelihood-tempered SMC imposes the joint distribution

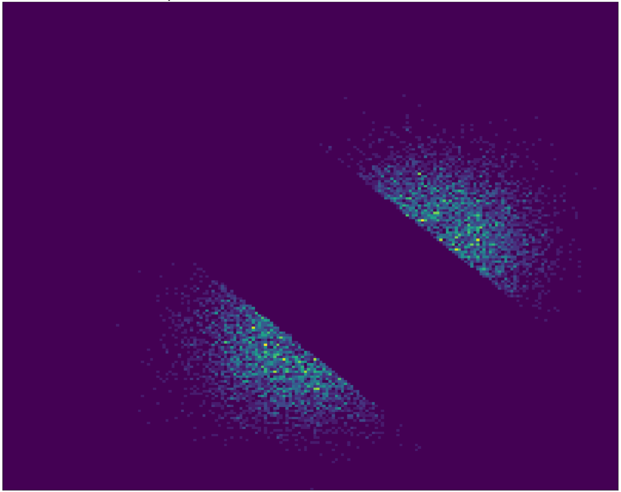
$$p(\theta_k, x) \propto \pi(\theta_k) p(x \mid \theta_k)^{\tau_k}.$$

- Temperatures satisfy $0 = \tau_1 < \cdots < \tau_T = 1$.

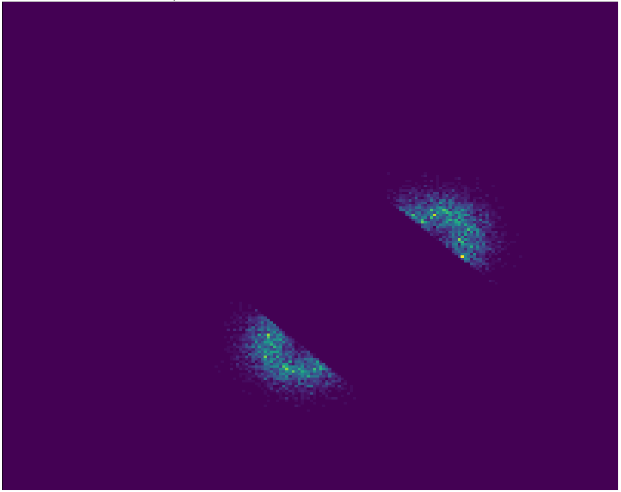
Temperature $\tau = 0.0$



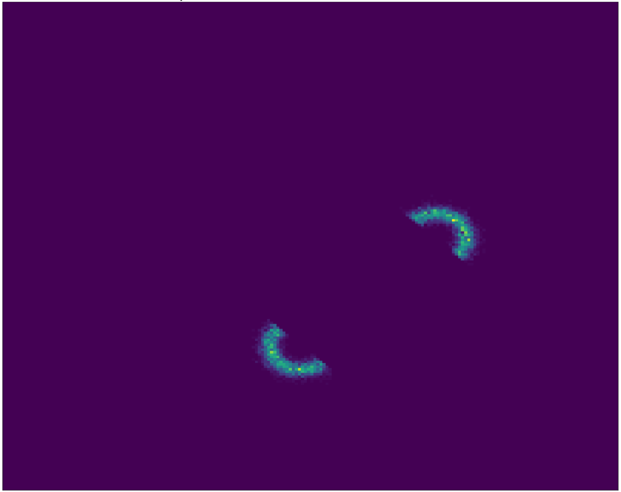
Temperature $\tau = 0.005289797269164934$



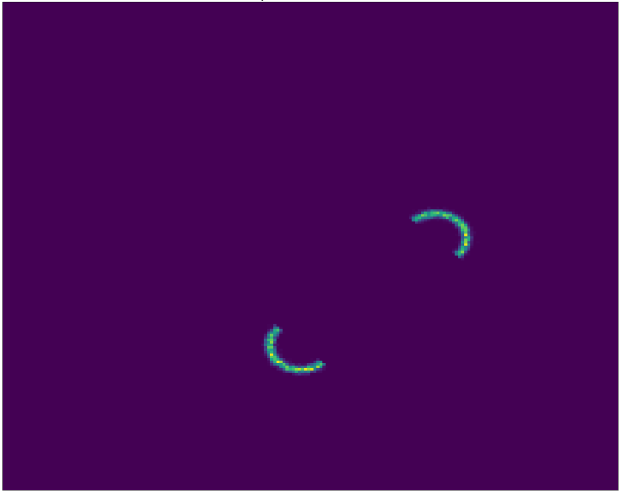
Temperature $\tau = 0.028516511264105 06$



Temperature $\tau = 0.2099392022875901$



Temperature $\tau = 1.0$



SMC-Wake

- Our proposed algorithm, SMC-Wake, trains the inference network using gradient updates

$$-\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{p(\theta|x_i)}\nabla_\phi \log q_\phi(\theta \mid x_i).$$

SMC-Wake

- Our proposed algorithm, SMC-Wake, trains the inference network using gradient updates

$$-\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{p(\theta|x_i)}\nabla_{\phi}\log q_{\phi}(\theta \mid x_i).$$

- Each expectation is estimated using particles sets constructed by SMC.

SMC-Wake

- Our proposed algorithm, SMC-Wake, trains the inference network using gradient updates

$$-\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{p(\theta|x_i)}\nabla_{\phi}\log q_{\phi}(\theta \mid x_i).$$

- Each expectation is estimated using particles sets constructed by SMC.
- Gradients estimated in this way are *consistent* as the number of particles $K \to \infty$.

Convergence

Lemma (Del Moral, 2004)

Let K be the number of particles used by likelihood-tempered SMC, and $\hat{P}(\theta_T)$ the empirical distribution at the final step. Then as $K \to \infty$,

$$\mathrm{KL}(\mathbb{E}\left[\hat{P}(\theta_T)\right] \;||\; p(\theta \mid x)) \to 0$$

Consistent Gradient Estimation

Theorem

Let

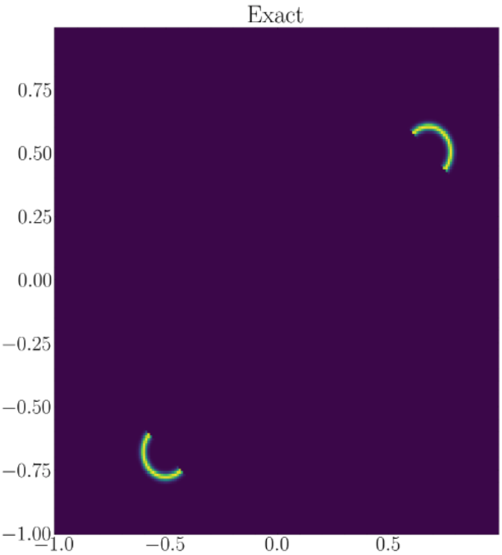
$$\psi = \mathbb{E}_{p(\theta|x)} \nabla_\phi \log q_\phi(\theta \mid x),$$
$$\hat{\psi} = \mathbb{E}_{\hat{P}(\theta_T)} \nabla_\phi \log q_\phi(\theta \mid x),$$

where $\hat{P}$ is a (random) discrete distribution resulting from an instance of SMC-Wake. Suppose that $|\nabla_\phi \log q_\phi(\theta \mid x)| \leq M$ (in each dimension), and that $\mathbb{E}(\hat{P}(\theta_T))$ is absolutely continuous with respect to $p(\theta \mid x)$. Then $\mathbb{E}(\hat{\psi}) - \psi \to 0$ as $K \to \infty$.

Experiment – Two Moons

$$
\begin{aligned}
\theta_1, \theta_2 &\overset{iid}{\sim} U(-1, 1), \\
a &\sim U\left(-\frac{\pi}{2}, \frac{\pi}{2}\right), \\
r &\sim \mathcal{TN}\left(0.1, 0.01^2; 0\right), \\
p &= (r\cos(a) + 0.25, r\sin(a)), \\
\text{and } x^\top &= p + \left(-\frac{|\theta_1 + \theta_2|}{\sqrt{2}}, \frac{-\theta_1 + \theta_2}{\sqrt{2}}\right).
\end{aligned}
$$

Two Moons Posterior



Experiments – Two Moons

- We sample 100 iid data points from the “two-moons” model.

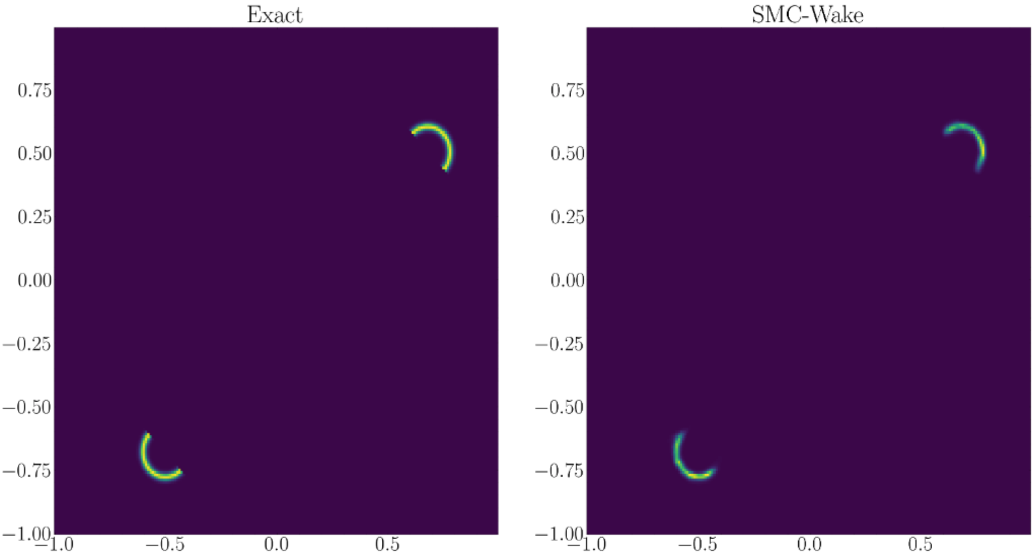
Experiments – Two Moons

- We sample 100 iid data points from the “two-moons” model.
- We train an inference network using SMC-Wake.

Experiments – Two Moons

- We sample 100 iid data points from the “two-moons” model.
- We train an inference network using SMC-Wake.
- We use the class Neural Spline Flows (NSP) as the variational family $\mathcal{Q}$.

Results – SMC-Wake



Competing Methods

Wake-phase training

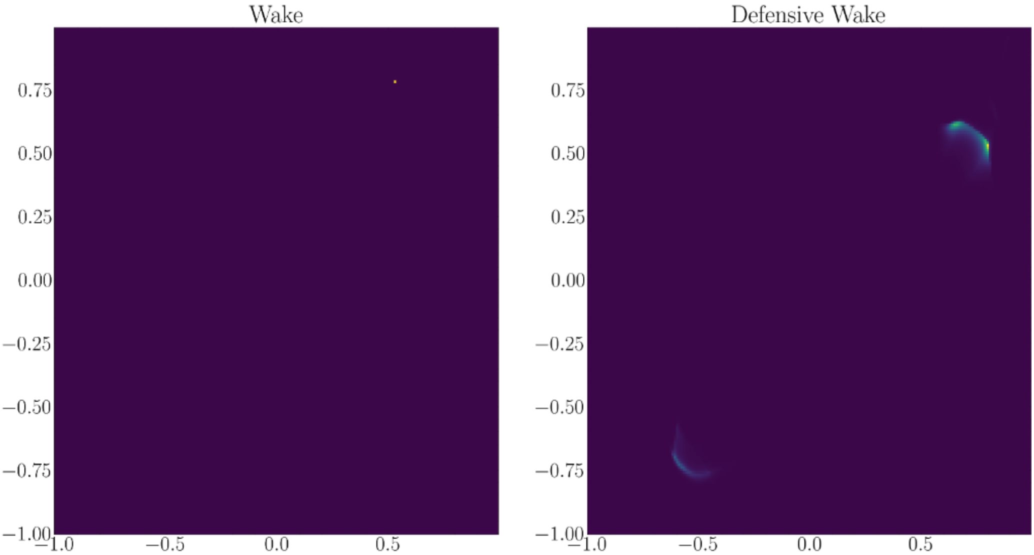
- Uses self-normalized importance sampling with q_ϕ as a proposal to estimate the gradient

$$-\mathbb{E}_{p(\theta|x)} \nabla_\phi \log q_\phi(\theta \mid x)$$

Defensive-Wake

- A variant that proposes with equal probability from either q_ϕ or the prior for importance sampling.

Results – Competitors



Discussion

SMC-Wake improves on wake-phase training by

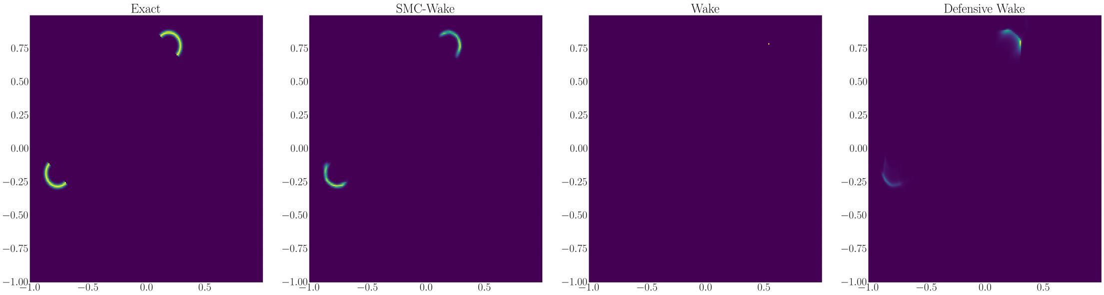
- Removing the pathology of proposing from and optimizing q_ϕ.

Discussion

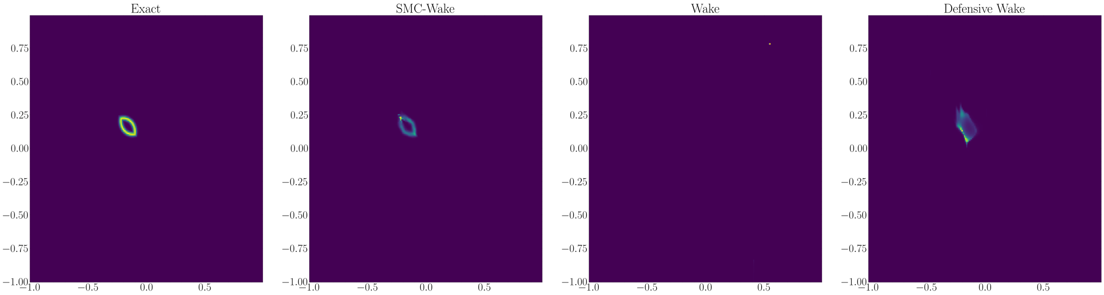
SMC-Wake improves on wake-phase training by

- Removing the pathology of proposing from and optimizing q_ϕ.
- Utilizing the likelihood more effectively via tempering to construct high-quality particle approximations.

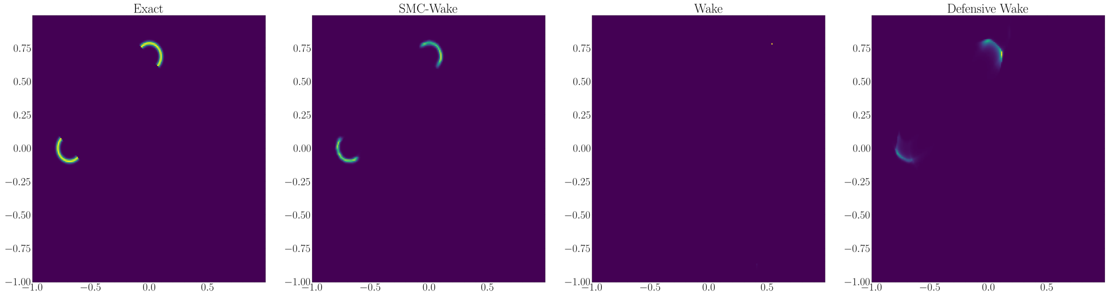
Additional Results



Additional Results



Additional Results



PROVABGS

- The Probabilisitc Value-Added Bright Galaxy Survey (PROVABGS) simulator is a forward model of galaxy spectra using 12 parameters.

PROVABGS

- The Probabilisitc Value-Added Bright Galaxy Survey (PROVABGS) simulator is a forward model of galaxy spectra using 12 parameters.
- We simulate from this forward model by sampling from a prior on the parameters, and subsequently
  - Add realistic noise.
  - Normalize out a magnitude parameter.

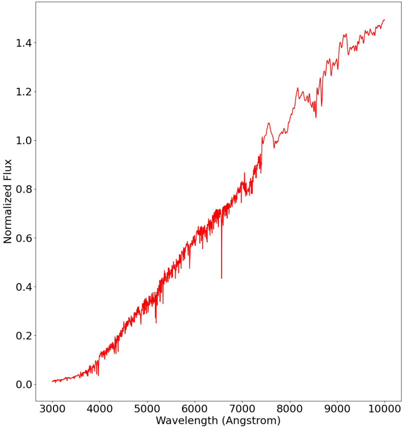
Inference For Galaxy Spectra

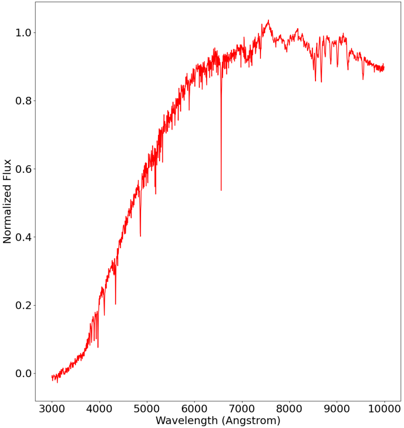
- Our PROVABGS emulator thus generates galaxy spectra from 11 distinct cosmological parameters.

Inference For Galaxy Spectra

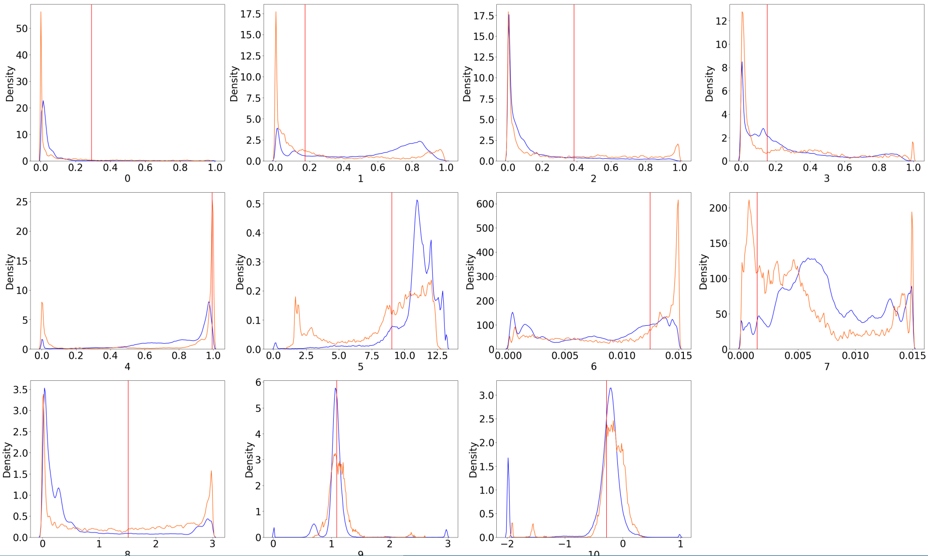
- Our PROVABGS emulator thus generates galaxy spectra from 11 distinct cosmological parameters.
- We add Gaussian noise to the emulator outputs, and aim to perform inference over these parameters using SMC-Wake.

PROVABGS

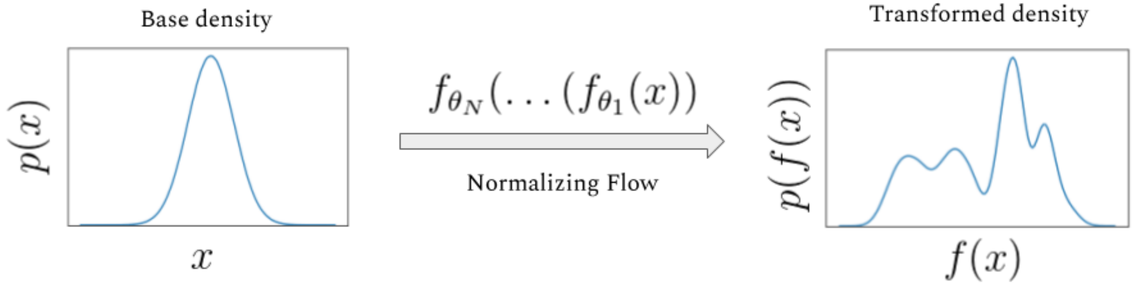

Normalized Flux
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
3000
4000
5000
6000
7000
8000
9000
10000
Wavelength (Angstrom)


Normalized Flux
0.0
0.2
0.4
0.6
0.8
1.0
3000
4000
5000
6000
7000
8000
9000
10000
Wavelength (Angstrom)

Results



Normalizing Flows



Source: https://gebob19.github.io/normalizing-flows/