Markovian Score Climbing

Variational Inference with KL(p||q)

Colab |Paper|GitHub

Christian Naesseth, Fredrik Lindsten, David Blei

Presented by Kelvin Wong

A Common Problem

- Consider a probabilistic model $p(z, x)$ over latent variables z and observed data x
- We are interested in computing the posterior distribution

$$p(z|x) = \frac{p(z, x)}{p(x)} = \frac{p(z, x)}{\int p(z, x) \mathrm{d}z}$$

- This is often intractable to compute!
- Solution: variational inference

Variational Inference

- Cast approximate inference as an optimization problem

$$q^\star(\cdot) = \operatorname*{argmin}_{q \in Q} D\big(p(\cdot\,|x), q(\cdot)\big)$$

- The solution $q^\star(\cdot)$ can be used as a surrogate to $p(\cdot\,|x)$

- Typically, the exclusive KL divergence is used

$$\mathrm{KL}(q(\cdot)||p(\cdot\,|x)) = \int q(z)\log\frac{q(z)}{p(z|x)}\,\mathrm{d}z$$

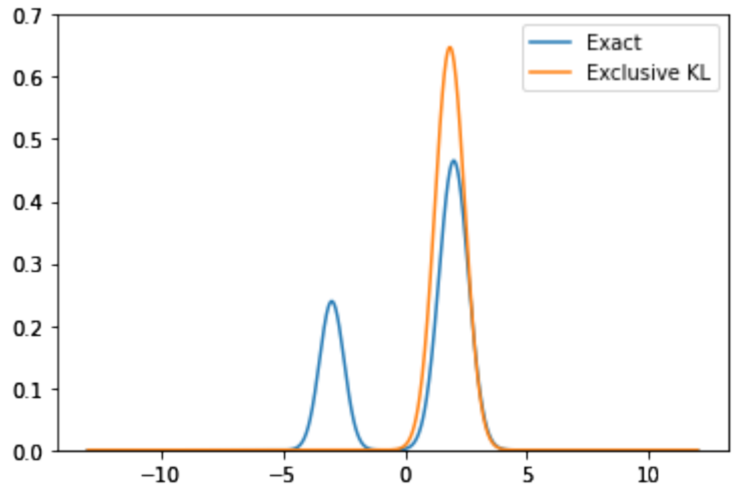
- This choice yields a convenient optimization problem; e.g. SGD

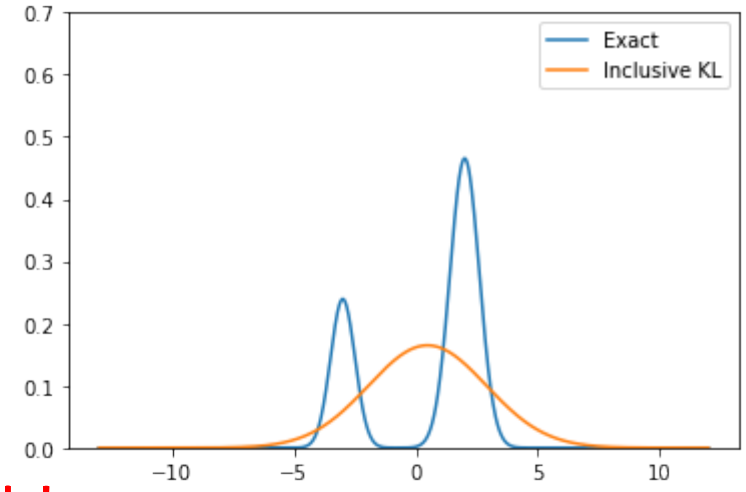
Inclusive vs. Exclusive KL Divergence

- Consider the following toy problem:
  - Given a bimodal Gaussian mixture model $p(z)$, find the Gaussian distribution $q(z; \mu, \sigma^2)$ that best approximates it.
- If we use the exclusive KL divergence to measure fit, then VI will yield a *mode-seeking* behavior.
- If we use the *inclusive KL divergence* to measure fit, then VI will yield a *mode-covering* behavior.

$$\mathrm{KL}(p(\cdot\,|x)||q(\cdot)) = \int p(z|x) \log \frac{p(z|x)}{q(z)} \mathrm{d}z$$

If this is 0, the KL may explode!





Contribution

- A method for minimizing the inclusive KL divergence using SGD
  - Prior methods use high variance or biased gradient estimates
  - In contrast, this method provably converges to a local minima
- **Key idea:** Use MCMC to estimate gradients for SGD
- This is called *Markovian Score Climbing*

Optimizing the Inclusive KL with SGD

- We want to minimize the following objective with SGD

$$\min_{\lambda} \int p(z|x) \log \frac{p(z|x)}{q(z;\lambda)} \mathrm{d}z$$

- The gradient of this objective is

$$g_{\mathrm{KL}}(\lambda) = -\mathrm{E}_{z \sim p(z|x)}[\nabla \log q(z;\lambda)]$$

- If we can estimate $g_{\mathrm{KL}}(\lambda)$, we can just apply SGD!
- Unfortunately, we don't know $p(z|x)$

Importance Sampling (IS)

- Re-write the expectation in terms of $q(z;\lambda)$

$$g_{\mathrm{KL}}(\lambda) = -\frac{1}{p(x)} \mathrm{E}_{z\sim q(z;\lambda)}\left[\frac{p(z,x)}{q(z;\lambda)} \nabla \log q(z;\lambda)\right]$$

$$\propto -\mathrm{E}_{z\sim q(z;\lambda)}\left[\frac{p(z,x)}{q(z;\lambda)} \nabla \log q(z;\lambda)\right]$$

- Estimate $g_{\mathrm{KL}}(\lambda)$ with Monte-Carlo estimation

$$\hat{g}_{\mathrm{KL}}(\lambda) = -\frac{1}{m}\sum_{i=1}^{m} \frac{p(z_i,x)}{q(z_i;\lambda)} \nabla \log q(z_i;\lambda)$$

- Unbiased but high variance

Self-normalized Importance Sampling (SIS)

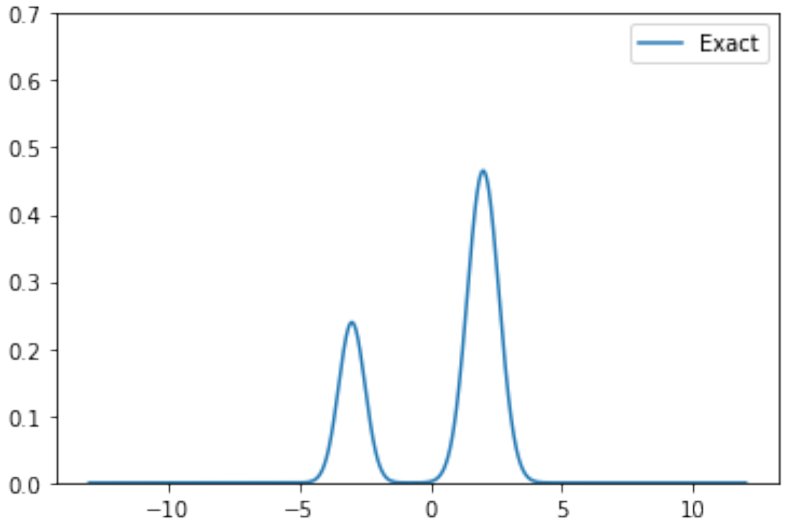
- Normalize importance weights to trade-off bias for variance

$$\hat{g}_{\mathrm{KL}}(\lambda) = -\frac{\sum_{i=1}^{m} \frac{p(z_i, x)}{q(z_i; \lambda)} \nabla \log q(z_i; \lambda)}{\sum_{i=1}^{m} \frac{p(z_i, x)}{q(z_i; \lambda)}}$$

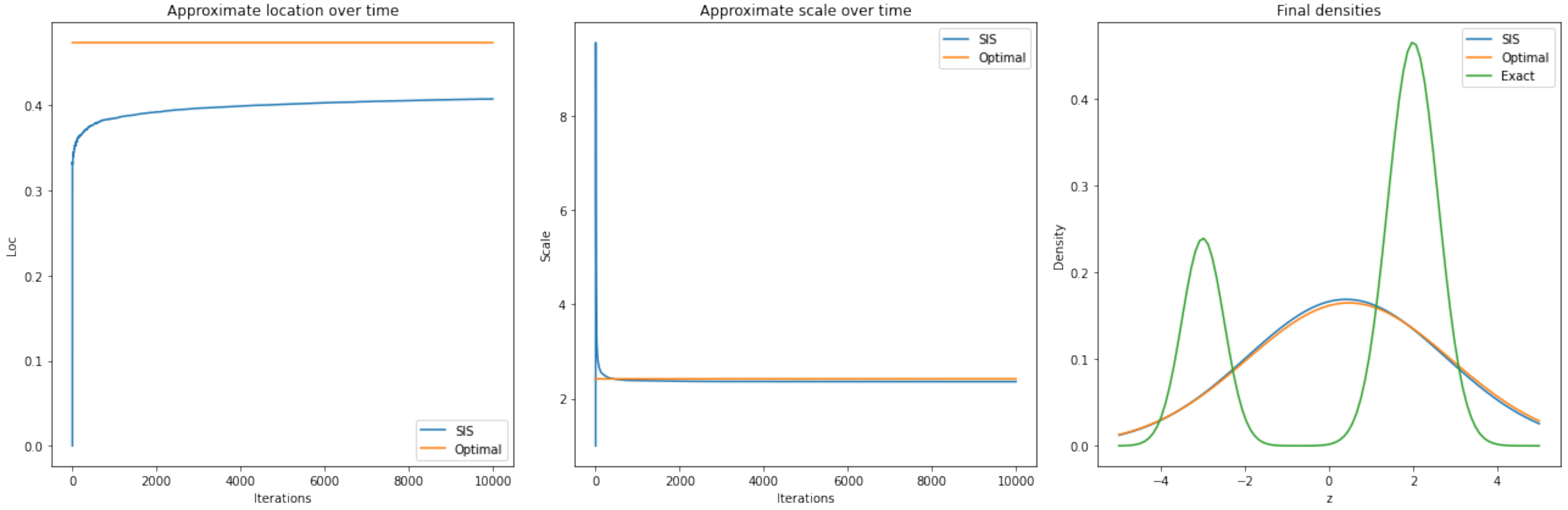
- Biased but lower variance
- It is also *consistent;* i.e., $\mathrm{E}[\hat{g}_{\mathrm{KL}}(\lambda)] \propto g_{\mathrm{KL}}(\lambda)$ when $m \to \infty$

Toy Example: SIS Estimator

- Recall our toy problem:
  - Given a bimodal Gaussian mixture model $p(z)$, find the Gaussian distribution $q(z; \mu, \sigma^2)$ that best approximates it.
- For each iteration of SGD
  - Sample $z_1, \ldots, z_m \sim q(z; \lambda_{k-1})$
  - Compute the SIS gradient estimate $\hat{g}_{\mathrm{KL}}(\lambda_{k-1})$
  - Run SGD $\lambda_k \leftarrow \lambda_{k-1} - \epsilon_k \hat{g}_{\mathrm{KL}}(\lambda_{k-1})$



Toy Example: SIS Estimator (cont.)



Markovian Score Climbing (MSC)

- **Key idea:** Use MCMC to estimate gradients for SGD
- For each iteration of SGD
  - Sample $z[k] \sim p(z|x)$ using MCMC
  - Compute $\hat{g}_{\mathrm{KL}}(\lambda_{k-1}) = -\nabla \log q(z[k]; \lambda_{k-1})$
  - Run SGD $\lambda_k \leftarrow \lambda_{k-1} - \epsilon_k \hat{g}_{\mathrm{KL}}(\lambda_{k-1})$
- We do not re-initialize the Markov chain at each iteration of SGD
- Under certain technical conditions, MSC provably converges to a local minima

MCMC in a Nutshell

- Class of algorithms to sample from an arbitrary distribution whose density is known proportionally
  - Build a Markov chain whose stationary distribution is $p(z|x)$
  - Starting from $z[0]$, traverse the chain until steady state
  - Output new states as samples $z[1], \dots, z[m]$
- The key is to design the Markov chain; i.e. the transition kernel

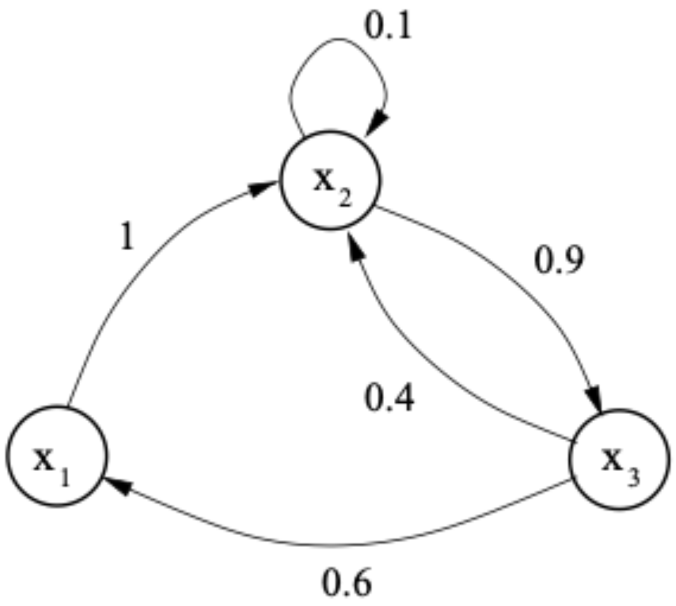
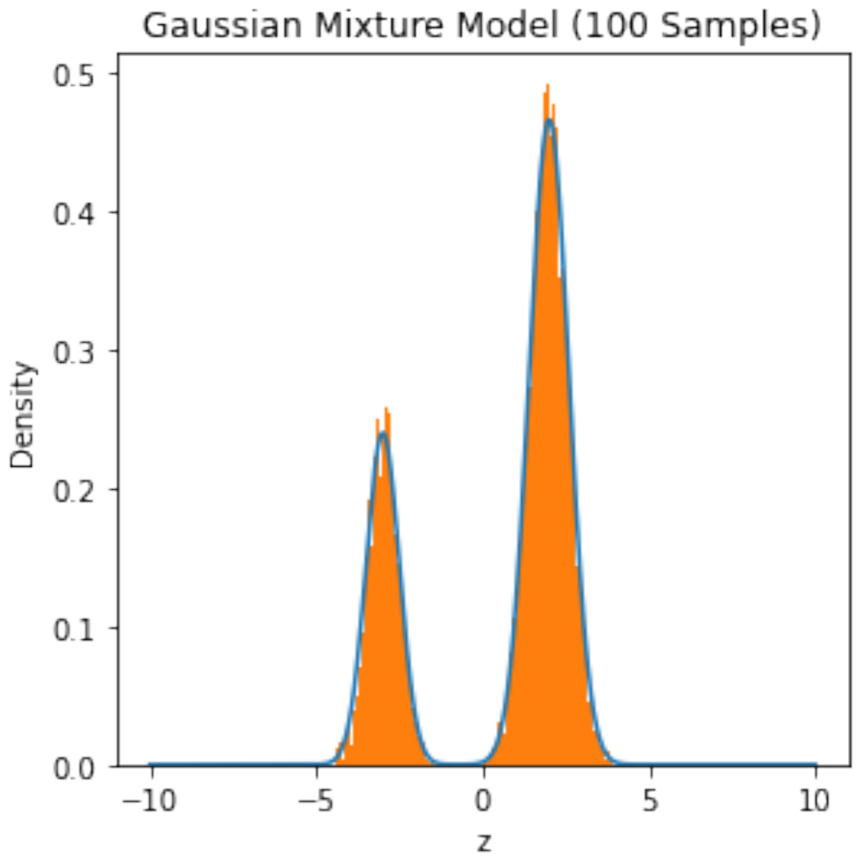


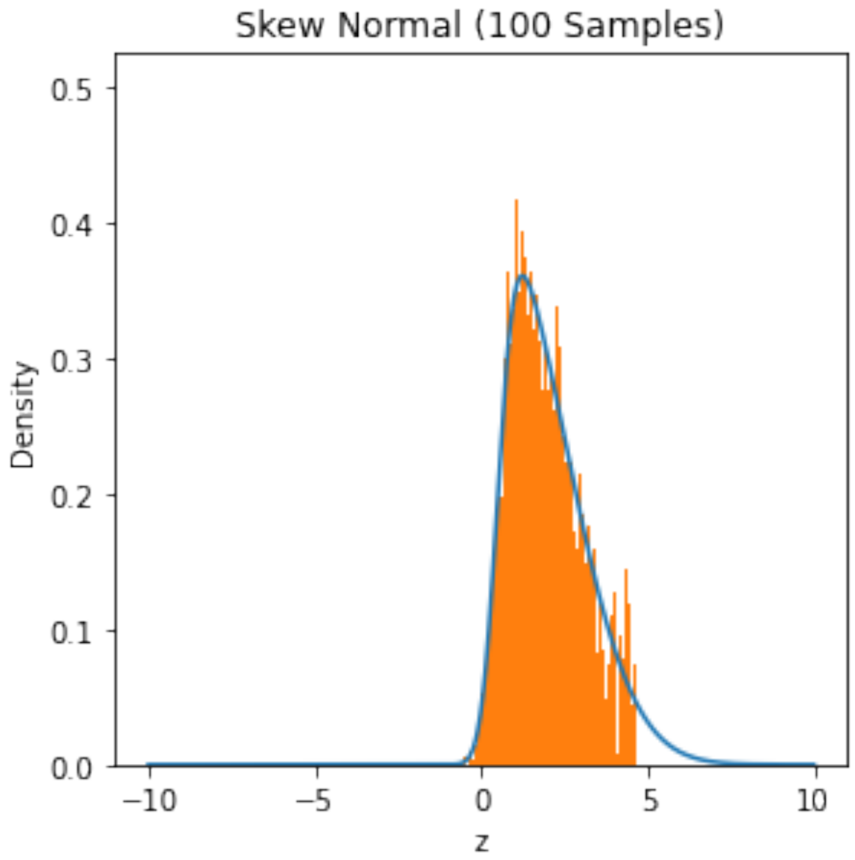
Figure from: Andrieu, De Freitas, Doucet, and Jordan. An Introduction to MCMC for Machine Learning

Conditional Importance Sampling (CIS)

- SIS-based Markov kernel with $p(z|x)$ as its stationary distribution
- For each iteration of SGD
  - Set $z_1 = z[k\ -1]$ and sample $z_2, \ldots, z_m \sim q(z; \lambda_{k-1})$
  - Compute self-normalized importance weights $w_i\ \propto \frac{p(z_i, x)}{q(z_i; \lambda_{k-1})}$
  - Sample $z[k\ -1]$ from $z_1, \ldots, z_m$ with proportional to $w_1, \ldots, w_m$

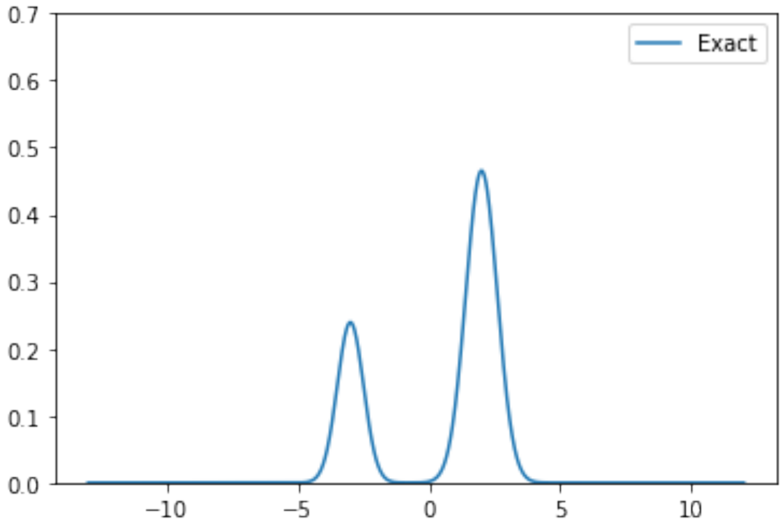
Example: Conditional Importance Sampling





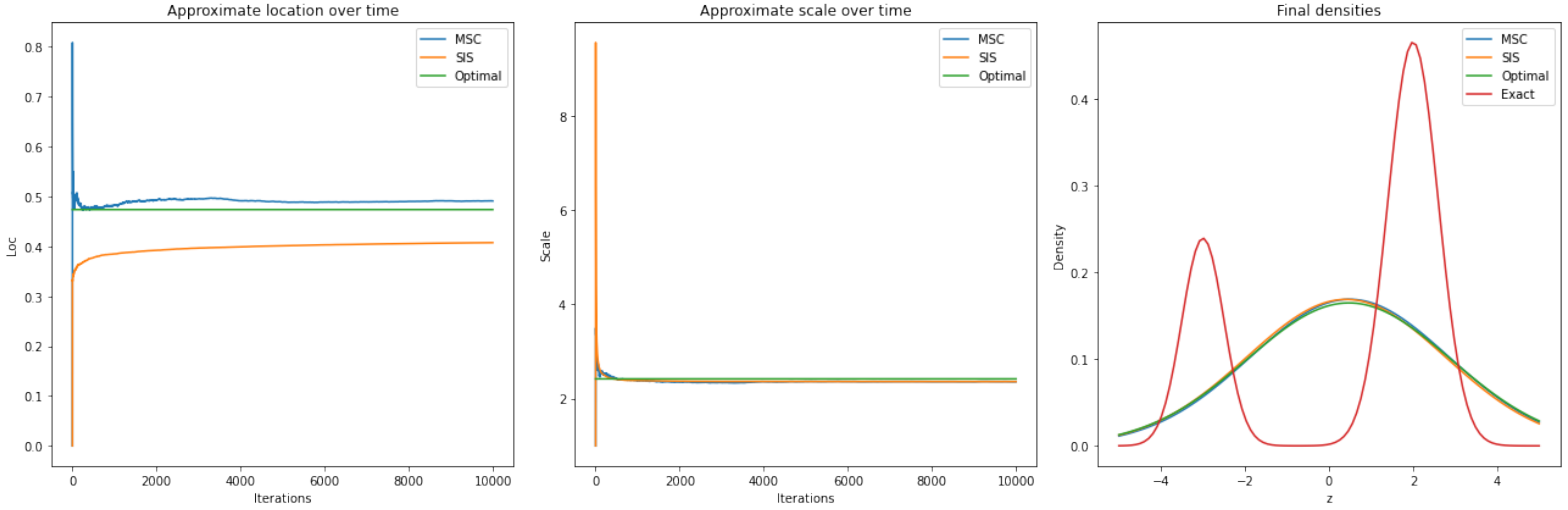
Putting Everything Together

- Recall our toy problem:
  - Given a bimodal Gaussian mixture model $p(z)$, find the Gaussian distribution $q(z; \mu, \sigma^2)$ that best approximates it.

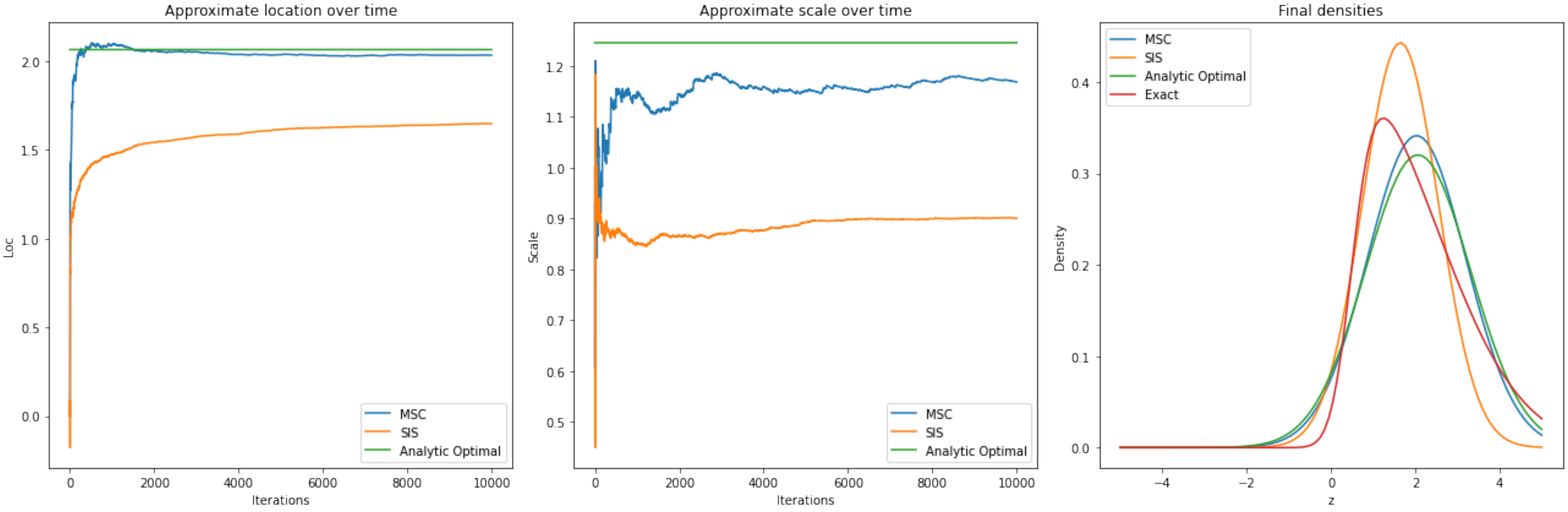


- For each iteration of SGD
  - Sample $z[k] \sim M(\cdot|z[k-1]; \lambda_{k-1})$ using CIS
  - Compute $\hat{g}_{\mathrm{KL}}(\lambda_{k-1}) = -\nabla \log q(z[k]; \lambda_{k-1})$
  - Run SGD $\lambda_k \leftarrow \lambda_{k-1} - \epsilon_k \hat{g}_{\mathrm{KL}}(\lambda_{k-1})$

Putting Everything Together (cont.)



Putting Everything Together (cont.)



Extension: Maximum Likelihood Estimation

- Suppose our probabilistic model has unknown parameters $p(z, x; \theta)$
- To fit the unknown parameters using maximum likelihood
  - Sample $z[k] \sim M(\cdot|z[k-1]; \lambda_{k-1})$ using CIS
  - Compute $\hat{g}_{\mathrm{KL}}(\lambda_{k-1}) = -\nabla \log q(z[k]; \lambda_{k-1})$
  - Run SGD $\lambda_k \leftarrow \lambda_{k-1} - \epsilon_k \hat{g}_{\mathrm{KL}}(\lambda_{k-1})$
  - Compute $\hat{g}_{\mathrm{ML}}(\theta_{k-1}) = -\nabla \log p(z[k], x; \theta_{k-1})$
  - Run SGD $\theta_k \leftarrow \theta_{k-1} - \epsilon_k \hat{g}_{\mathrm{ML}}(\theta_{k-1})$

Other Extensions

- Extension to state-space models using Sequential Monte Carlo (SMC)
  - Key idea: Replace CIS with conditional SMC (CSMC)
  - At each iteration, resample $m-1$ particles and set the retained particle as the m-th one
- Extension to large-scale datasets
  - If the observed data $x_1, \dots, x_m$ are IID, consider minimizing the expected inclusive KL instead

$$\min_{q \in Q} \mathrm{E}_{x_i \sim p(x_i)}\left[\mathrm{KL}\left(p(\cdot \mid x_i) \| q(\cdot)\right)\right]$$

  - Gradients can be estimated as

$$\hat{g}_{\mathrm{KL}}(\lambda) = -\frac{1}{m} \sum_{i=1}^{m} \mathrm{E}_{z \sim p(z \mid x_i)}\left[\nabla \log q\left(z; \lambda(x_i)\right)\right]$$

Related Work

- Other variational objectives
  - Renyi α-divergences (Li and Turner, 2016)
  - (Langevin-Stein) operator variational objective (Ranganath et al., 2016)
  - χ-divergences (Dieng et al., 2017)
  - Thermodynamic variational objectives (Masrani et al., 2019)
  - Variational contrastive divergence (Ruiz and Titsias, 2019)
- Other work minimizing the inclusive KL
  - Expectation propagation (Minka, 2001)
  - Reweighted Wake-Sleep (Bornschein and Bengio, 2015)
  - Neural Adaptive Sequential Monte Carlo (Gu et al., 2015)

Discussion

- Markovian Score Climbing for minimizing the inclusive KL divergence using SGD
- **Key idea:** Use MCMC to estimate gradients for SGD
- But:
  - Applications to large-scale datasets has not been explored
  - Conditions for convergence is difficult to verify in general
  - Unclear how fast MSC converges in practice
- Questions?