Stochastic Gradient MCMC for Nonlinear State Space Models

Srshti Putcha

Joint work with:
Chris Aicher, Chris Nemeth, Paul Fearnhead, and Emily Fox

Problem Motivation

Figure 1: Exchange Rate Data

- Conduct Bayesian inference for long sequences of time series data

Problem Motivation

Figure 1: Exchange Rate Data

- Conduct Bayesian inference for long sequences of time series data
- Focus on nonlinear, non-Gaussian **state space models** (SSMs)

Background: Notation

State Space Models

a) **Hidden** or **latent** process, $\{X_t\}_{t \geq 1}$,

Background: Notation

State Space Models

a) **Hidden** or **latent** process, $\{X_t\}_{t \geq 1}$,

- initial prior, $X_0 \sim \nu(x_0|\theta)$,

Background: Notation

State Space Models

a) **Hidden** or **latent** process, $\{X_t\}_{t\geq 1}$,

- initial prior, $X_0 \sim \nu(x_0|\theta)$,
- transition density, $X_t|(X_{t-1} = x_{t-1}), \theta \sim p(x_t|x_{t-1}, \theta)$.

Background: Notation

State Space Models

a) **Hidden** or **latent** process, $\{X_t\}_{t \geq 1}$,

- initial prior, $X_0 \sim \nu(x_0|\theta)$,
- transition density, $X_t|(X_{t-1} = x_{t-1}), \theta \sim p(x_t|x_{t-1}, \theta)$.

b) **Observed** process, $\{Y_t\}_{t \geq 1}$,

Background: Notation

State Space Models

a) **Hidden** or **latent** process, $\{X_t\}_{t\geq 1}$,

- initial prior, $X_0 \sim \nu(x_0|\theta)$,
- transition density, $X_t|(X_{t-1} = x_{t-1}), \theta \sim p(x_t|x_{t-1}, \theta)$.

b) **Observed** process, $\{Y_t\}_{t\geq 1}$,

- emission density, $Y_t|(X_t = x_t), \theta \sim p(y_t|x_t, \theta)$.

Background: Notation

State Space Models

a) **Hidden** or **latent** process, $\{X_t\}_{t\geq 1}$,

- initial prior, $X_0 \sim \nu(x_0|\theta)$,
- transition density, $X_t|(X_{t-1} = x_{t-1}), \theta \sim p(x_t|x_{t-1}, \theta)$.

b) **Observed** process, $\{Y_t\}_{t\geq 1}$,

- emission density, $Y_t|(X_t = x_t), \theta \sim p(y_t|x_t, \theta)$.

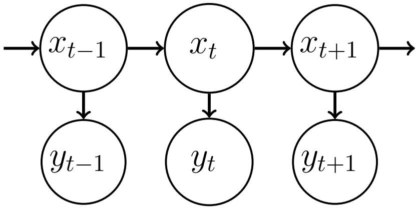

Figure 2: State Space Model

Background: Fisher's Identity

Background: Fisher's Identity

- Complete data loglikelihood, $p(y_{1:T}, x_{1:T}, \theta)$.

Background: Fisher's Identity

- Complete data loglikelihood, $p(y_{1:T}, x_{1:T}, \theta)$.
- Latent state posterior, $p(x_{1:T} | y_{1:T}, \theta)$.

Background: Fisher's Identity

- Complete data loglikelihood, $p(y_{1:T}, x_{1:T}, \theta)$.
- Latent state posterior, $p(x_{1:T} | y_{1:T}, \theta)$.
- Score function, $\nabla_\theta \log p(y_{1:T} | \theta)$.

Background: Fisher's Identity

- Complete data loglikelihood, $p(y_{1:T}, x_{1:T}, \theta)$.
- Latent state posterior, $p(x_{1:T}|y_{1:T}, \theta)$.
- Score function, $\nabla_\theta \log p(y_{1:T}|\theta)$.

Fisher's Identity (Cappé et al., 2005)

$$\begin{aligned}\nabla_\theta \log p(y_{1:T}|\theta) &= \mathbb{E}_{X|Y,\theta}[\nabla_\theta \log p(y_{1:T}, X_{1:T}|\theta)] \\ &= \sum_{t=1}^{T} \mathbb{E}_{X|Y,\theta}[\nabla_\theta \log p(y_t, X_t|x_{t-1}, \theta)]\end{aligned}$$

Background: Fisher's Identity

- Complete data loglikelihood, $p(y_{1:T}, x_{1:T}, \theta)$.
- Latent state posterior, $p(x_{1:T}|y_{1:T}, \theta)$.
- Score function, $\nabla_\theta \log p(y_{1:T}|\theta)$.

Fisher's Identity (Cappé et al., 2005)

$$\begin{aligned}
\nabla_\theta \log p(y_{1:T}|\theta) &= \mathbb{E}_{X|Y,\theta}[\nabla_\theta \log p(y_{1:T}, X_{1:T}|\theta)] \\
&= \sum_{t=1}^{T} \mathbb{E}_{X|Y,\theta}[\nabla_\theta \log p(y_t, X_t|x_{t-1}, \theta)]
\end{aligned}$$

- What happens when we cannot express the latent state posterior in closed form?

Background: Particle Filtering

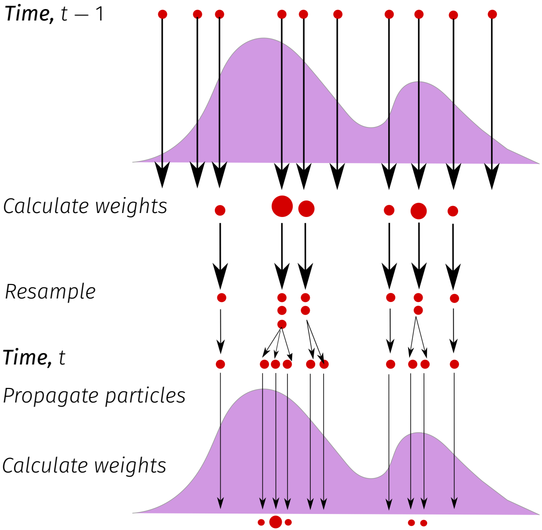

Figure 3: Sequential Importance Resampling (SIR)

Background: Particle Filtering

Assume a collection of N particles, $\{x_t^{(i)}\}_{i=1}^N$ and weights, $\{w_t^{(i)}\}_{i=1}^N$.

Background: Particle Filtering

Assume a collection of N particles, $\{x_t^{(i)}\}_{i=1}^N$ and weights, $\{w_t^{(i)}\}_{i=1}^N$.

Sequential Importance Resampling (Gordon et al., 1993)

1. **Resample** ancestor indices $\{a_1, \ldots, a_N\}$, i.e.
   $a_i \sim \text{Categorical}(w_{t-1}^{(i)})$.

Background: Particle Filtering

Assume a collection of N particles, $\{x_t^{(i)}\}_{i=1}^N$ and weights, $\{w_t^{(i)}\}_{i=1}^N$.

Sequential Importance Resampling (Gordon et al., 1993)

1. **Resample** ancestor indices $\{a_1, \ldots, a_N\}$, i.e. $a_i \sim \text{Categorical}(w_{t-1}^{(i)})$.
2. **Propagate** $x_t^{(i)} \sim q(\cdot | x_{t-1}^{(a_i)}, y_t, \theta)$, using a proposal distribution $q(\cdot|\cdot)$.

Background: Particle Filtering

Assume a collection of N particles, $\{x_t^{(i)}\}_{i=1}^N$ and weights, $\{w_t^{(i)}\}_{i=1}^N$.

Sequential Importance Resampling (Gordon et al., 1993)

1. **Resample** ancestor indices $\{a_1, \dots, a_N\}$, i.e. $a_i \sim \text{Categorical}(w_{t-1}^{(i)})$.
2. **Propagate** $x_t^{(i)} \sim q(\cdot|x_{t-1}^{(a_i)}, y_t, \theta)$, using a proposal distribution $q(\cdot|\cdot)$.
3. **Update** and normalize weights,

$$w_t^{(i)} \propto \frac{p(y_t|x_t^{(i)}, \theta)p(x_t^{(i)}|x_{t-1}^{(a_i)}, \theta)}{q(x_t^{(i)}|x_{t-1}^{(a_i)}, y_t, \theta)}, \quad \sum_i w_t^{(i)} = 1.$$

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of a function $H(X_{1:T})$ with respect to the latent state posterior.

Goal: create an empirical approximation of the expectation of a function $H(X_{1:T})$ with respect to the latent state posterior.

- If $H(x_{1:T}) = \sum_{t=1}^{T} h_t(x_t, x_{t-1})$, we only need to track the partial sums $H_t = \sum_{s=1}^{t} h_s(x_s, x_{s-1})$.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of a function $H(X_{1:T})$ with respect to the latent state posterior.

- If $H(x_{1:T}) = \sum_{t=1}^{T} h_t(x_t, x_{t-1})$, we only need to track the partial sums $H_t = \sum_{s=1}^{t} h_s(x_s, x_{s-1})$.
- Score approximation can be constructed by setting:

$$h_t(x_t, x_{t-1}) = \nabla_\theta \log p(y_t, X_t | x_{t-1}, \theta).$$

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of a function $H(X_{1:T})$ with respect to the latent state posterior.

- If $H(x_{1:T}) = \sum_{t=1}^{T} h_t(x_t, x_{t-1})$, we only need to track the partial sums $H_t = \sum_{s=1}^{t} h_s(x_s, x_{s-1})$.
- Score approximation can be constructed by setting:

$$h_t(x_t, x_{t-1}) = \nabla_\theta \log p(y_t, X_t | x_{t-1}, \theta).$$

- Poyiadjis et al. (2011), Nemeth et al. (2016), and Olsson et al. (2017) propose various score approximations of this form.

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Draw $x_0^{(i)} \sim \nu(x_0|\theta)$, set $w_0^{(i)} = \frac{1}{N}$, and $H_0^{(i)} = 0 \; \forall i$.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Draw $x_0^{(i)} \sim \nu(x_0|\theta)$, set $w_0^{(i)} = \frac{1}{N}$, and $H_0^{(i)} = 0 \ \forall i$.

For $t = 1, \ldots, T$,

1. Resample ancestor indices $\{a_1, \ldots, a_N\}$.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Draw $x_0^{(i)} \sim \nu(x_0|\theta)$, set $w_0^{(i)} = \frac{1}{N}$, and $H_0^{(i)} = 0 \; \forall i$.

For $t = 1, \ldots, T$,

1. Resample ancestor indices $\{a_1, \ldots, a_N\}$.
2. Propagate particles $x_t^{(i)} \sim q(\cdot|x_{t-1}^{(a_i)}, y_t, \theta)$.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Draw $x_0^{(i)} \sim \nu(x_0|\theta)$, set $w_0^{(i)} = \frac{1}{N}$, and $H_0^{(i)} = 0\ \forall i$.

For $t = 1, \ldots, T$,

1. Resample ancestor indices $\{a_1, \ldots, a_N\}$.
2. Propagate particles $x_t^{(i)} \sim q(\cdot|x_{t-1}^{(a_i)}, y_t, \theta)$.
3. Update each $w_t^{(i)}$.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Draw $x_0^{(i)} \sim \nu(x_0|\theta)$, set $w_0^{(i)} = \frac{1}{N}$, and $H_0^{(i)} = 0\ \forall i$.

For $t = 1, \ldots, T$,

1. Resample ancestor indices $\{a_1, \ldots, a_N\}$.
2. Propagate particles $x_t^{(i)} \sim q(\cdot|x_{t-1}^{(a_i)}, y_t, \theta)$.
3. Update each $w_t^{(i)}$.
4. Update statistics $H_t^{(i)} = H_{t-1}^{(a_i)} + h_t(x_t^{(i)}, x_{t-1}^{(a_i)})$.

Background: Particle Filtering

Goal: create an empirical approximation of the expectation of the **score function** with respect to the latent state posterior.

Algorithm 1 - Particle Filter

Input: number of particles, N, pairwise statistics, $h_{1:T}$, observations $y_{1:T}$, proposal density q.

Draw $x_0^{(i)} \sim \nu(x_0|\theta)$, set $w_0^{(i)} = \frac{1}{N}$, and $H_0^{(i)} = 0\ \forall i$.

For $t = 1, \ldots, T$,

1. Resample ancestor indices $\{a_1, \ldots, a_N\}$.
2. Propagate particles $x_t^{(i)} \sim q(\cdot|x_{t-1}^{(a_i)}, y_t, \theta)$.
3. Update each $w_t^{(i)}$.
4. Update statistics $H_t^{(i)} = H_{t-1}^{(a_i)} + h_t(x_t^{(i)}, x_{t-1}^{(a_i)})$.

Return $H = \sum_{i=1}^{N} w_T^{(i)} H_T^{(i)}$.

Background: Particle Filtering

- How do things change for large values of T?

Background: Particle Filtering

- How do things change for large values of T?
- Two-fold challenge for parameter inference:

Background: Particle Filtering

- How do things change for large values of T?
- Two-fold challenge for parameter inference:
    a) the cost of each pass of the data scales with the length of the sequence

Background: Particle Filtering

- How do things change for large values of T?
- Two-fold challenge for parameter inference:
  a) the cost of each pass of the data scales with the length of the sequence
  b) particle methods suffer from degeneracy

Background: Stochastic Gradient MCMC

Goal: draw a sample θ from the posterior $p(\theta|y) \propto p(\theta)p(y|\theta)$.

Background: Stochastic Gradient MCMC

Goal: draw a sample θ from the posterior $p(\theta|y) \propto p(\theta)p(y|\theta)$.

Idea: simulate from an SDE based on the gradient of the loglikelihood,

$$g_\theta = \nabla_\theta \log p(y|\theta).$$

Background: Stochastic Gradient MCMC

Goal: draw a sample θ from the posterior $p(\theta|y) \propto p(\theta)p(y|\theta)$.

Idea: simulate from an SDE based on the gradient of the loglikelihood,

$$g_\theta = \nabla_\theta \log p(y|\theta).$$

Method: construct a stochastic gradient, $\widehat{g}_\theta$, using subsamples of data.

Background: Stochastic Gradient MCMC

Goal: draw a sample θ from the posterior $p(\theta|y) \propto p(\theta)p(y|\theta)$.

Idea: simulate from an SDE based on the gradient of the loglikelihood,

$$g_\theta = \nabla_\theta \log p(y|\theta).$$

Method: construct a stochastic gradient, $\widehat{g}_\theta$, using subsamples of data.

SGLD Algorithm (Welling and Teh, 2011)

Input: initial $\theta^{(0)}$, stepsizes $\{\epsilon^{(k)}\}$, data y.
For $k = 1, 2, \ldots, K$,

$$\theta^{(k+1)} \leftarrow \theta^{(k)} + \epsilon^{(k)} \cdot \widehat{g}_\theta + \mathcal{N}(0, 2\epsilon^{(k)}).$$

Method: Overview

Goal: apply SGLD to nonlinear, non-Gaussian SSMs.

Method: Overview

Goal: apply SGLD to nonlinear, non-Gaussian SSMs.

Idea: use a **buffered** stochastic gradient

Method: Overview

Goal: apply SGLD to nonlinear, non-Gaussian SSMs.

Idea: use a **buffered** stochastic gradient

Buffered Stochastic Gradient (Aicher et al., 2018)

Consider a contiguous subsequence of length S,
$\mathcal{S} = \{s+1, \ldots, s+S\}$. We propose:

$$\widehat{g}_\theta(S, B) = \sum_{t \in \mathcal{S}} \frac{\mathbb{E}_{x|y^*_{\mathcal{S}},\theta}[\nabla_\theta \log p(X_t, y_t | x_{t-1}, \theta)]}{\Pr(t \in \mathcal{S})},$$

where B is the buffer length and $\mathcal{S}^*\{s+1-B, \ldots, s+S+B\}$.

Method: Overview

Goal: apply SGLD to nonlinear, non-Gaussian SSMs.

Idea: use a **buffered** stochastic gradient

Buffered Stochastic Gradient (Aicher et al., 2018)

Consider a contiguous subsequence of length S, $\mathcal{S} = \{s+1, \ldots, s+S\}$. We propose:

$$\widehat{g}_\theta(S, B) = \sum_{t \in \mathcal{S}} \frac{\mathbb{E}_{x|y^*_{\mathcal{S}},\theta}[\nabla_\theta \log p(X_t, y_t | x_{t-1}, \theta)]}{\Pr(t \in \mathcal{S})},$$

where B is the buffer length and $\mathcal{S}^*\{s+1-B, \ldots, s+S+B\}$.

Note: If the SSM and its gradient satisfy a Lipschitz condition, the bias decays geometrically in B.

Method: Overview

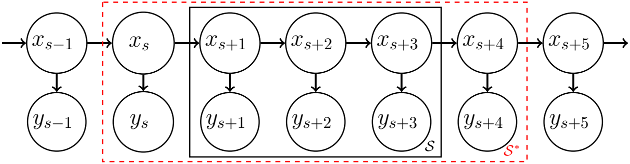

Figure 4: Graphical model of $\mathcal{S}^*$ with $S = 3$ and $B = 2$

Method: Overview

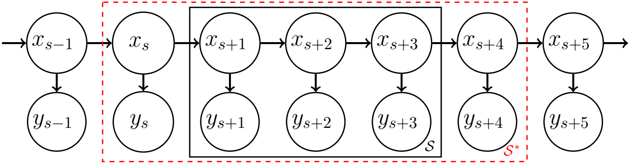

Figure 4: Graphical model of $\mathcal{S}^*$ with $S = 3$ and $B = 2$

Method: construct a particle appoximation of $\widehat{g}_\theta(S, B)$ suitable for nonlinear SSMs, $g_\theta^{PF}(S, B, N)$.

Method: Overview

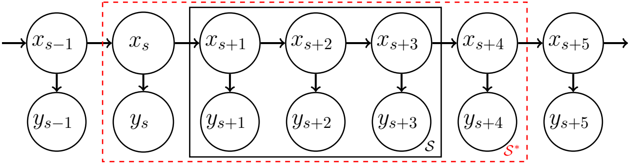

Figure 4: Graphical model of $\mathcal{S}^*$ with $S = 3$ and $B = 2$

Method: construct a particle appoximation of $\widehat{g}_\theta(S, B)$ suitable for nonlinear SSMs, $g_\theta^{PF}(S, B, N)$.

We decompose the complete data loglikelihood, $p(y_\mathcal{S}, x_\mathcal{S}|\theta)$ into the sum, $H = \sum_{t \in \mathcal{S}^*} h_t(x_t, x_{t-1})$,

$$h_t(x_t, x_{t-1}) = \begin{cases} \dfrac{\nabla_\theta \log p(x_t, y_t \mid x_{t-1}, \theta)}{\Pr(t \in \mathcal{S}} & \text{if } t \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

Algorithm 2 - Buffered PF-SGLD (Aicher et al., 2019)

Input: data $y_{1:T}$, initial $\theta^{(0)}$, stepsize ϵ, subsequence size S, buffer size B, particle size N.

Method: SGMCMC Algorithm

Algorithm 2 - Buffered PF-SGLD (Aicher et al., 2019)

Input: data $y_{1:T}$, initial $\theta^{(0)}$, stepsize ϵ, subsequence size S, buffer size B, particle size N.

For $k = 1, 2, \ldots, K$,

1. Sample $\mathcal{S} = \{s + 1, \ldots, s + S\}$,

Method: SGMCMC Algorithm

Algorithm 2 - Buffered PF-SGLD (Aicher et al., 2019)

Input: data $y_{1:T}$, initial $\theta^{(0)}$, stepsize ϵ, subsequence size S, buffer size B, particle size N.

For $k = 1, 2, \ldots, K$,

1. Sample $\mathcal{S} = \{s + 1, \ldots, s + S\}$,
2. Set $\mathcal{S}^* = \{s + 1 - B, \ldots, s + S + B\}$,

Algorithm 2 - Buffered PF-SGLD (Aicher et al., 2019)

Input: data $y_{1:T}$, initial $\theta^{(0)}$, stepsize ϵ, subsequence size S, buffer size B, particle size N.

For $k = 1, 2, \ldots, K$,

1. Sample $\mathcal{S} = \{s+1, \ldots, s+S\}$,
2. Set $\mathcal{S}^* = \{s+1-B, \ldots, s+S+B\}$,
3. Calculate g_θ^{PF},

Method: SGMCMC Algorithm

Algorithm 2 - Buffered PF-SGLD (Aicher et al., 2019)

Input: data $y_{1:T}$, initial $\theta^{(0)}$, stepsize ϵ, subsequence size S, buffer size B, particle size N.

For $k = 1, 2, \ldots, K$,

1. Sample $\mathcal{S} = \{s + 1, \ldots, s + S\}$,
2. Set $\mathcal{S}^* = \{s + 1 - B, \ldots, s + S + B\}$,
3. Calculate g_θ^{PF},
4. Set $\theta^{(k+1)} \leftarrow \theta^{(k)} + \epsilon \cdot g_\theta^{PF} + \mathcal{N}(0, 2\epsilon)$.

Method: SGMCMC Algorithm

Algorithm 2 - Buffered PF-SGLD (Aicher et al., 2019)

Input: data $y_{1:T}$, initial $\theta^{(0)}$, stepsize ϵ, subsequence size S, buffer size B, particle size N.
For $k = 1, 2, \ldots, K$,

1. Sample $\mathcal{S} = \{s+1, \ldots, s+S\}$,
2. Set $\mathcal{S}^* = \{s+1-B, \ldots, s+S+B\}$,
3. Calculate g_θ^{PF},
4. Set $\theta^{(k+1)} \leftarrow \theta^{(k)} + \epsilon \cdot g_\theta^{PF} + \mathcal{N}(0, 2\epsilon)$.

Return θ^{K+1}.

Models

Models

- Linear Gaussian SSM (LGSSM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid x_t\,,\, \tau^2).$$

Experiments

Models

- Linear Gaussian SSM (LGSSM)

$$X_t \,|\, (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \,|\, \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \,|\, (X_t = x_t), \theta \sim \mathcal{N}(y_t \,|\, x_t\,,\, \tau^2).$$

- Stochastic Volatility Model (SVM)

$$X_t \,|\, (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \,|\, \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \,|\, (X_t = x_t), \theta \sim \mathcal{N}(y_t \,|\, 0\,,\, \exp(x_t)\tau^2).$$

Experiments

Models

- Linear Gaussian SSM (LGSSM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid x_t\,,\, \tau^2).$$

- Stochastic Volatility Model (SVM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid 0\,,\, \exp(x_t)\tau^2).$$

Evaluation Metrics

Experiments

Models

- Linear Gaussian SSM (LGSSM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid x_t\,,\, \tau^2).$$

- Stochastic Volatility Model (SVM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid 0\,,\, \exp(x_t)\tau^2).$$

Evaluation Metrics

- MSE of estimated posterior mean

Experiments

Models

- Linear Gaussian SSM (LGSSM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid x_t\,,\, \tau^2).$$

- Stochastic Volatility Model (SVM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid 0\,,\, \exp(x_t)\tau^2).$$

Evaluation Metrics

- MSE of estimated posterior mean
- Heldout loglikelihood

Models

- Linear Gaussian SSM (LGSSM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid x_t\,,\, \tau^2).$$

- Stochastic Volatility Model (SVM)

$$X_t \mid (X_{t-1} = x_{t-1}), \theta \sim \mathcal{N}(x_t \mid \phi x_{t-1}\,,\, \sigma^2),$$
$$Y_t \mid (X_t = x_t), \theta \sim \mathcal{N}(y_t \mid 0\,,\, \exp(x_t)\tau^2).$$

Evaluation Metrics

- MSE of estimated posterior mean
- Heldout loglikelihood
- Predictive loglikelihood

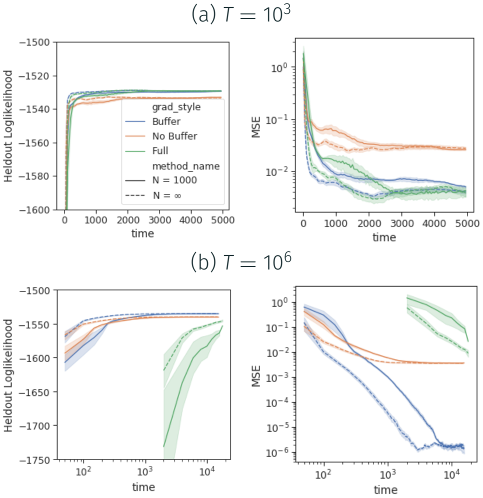

Figure 5: SGLD on Synthetic LGSSM data. (top) $T = 10^3$, (bottom) $T = 10^6$. (left) heldout-loglikelihood, (right) MSE of estimated posterior mean to true $\phi = 0.9$.

SGLD on Exchange-Rate Data

We now consider fitting the SVM to EUR-US exchange rate data.

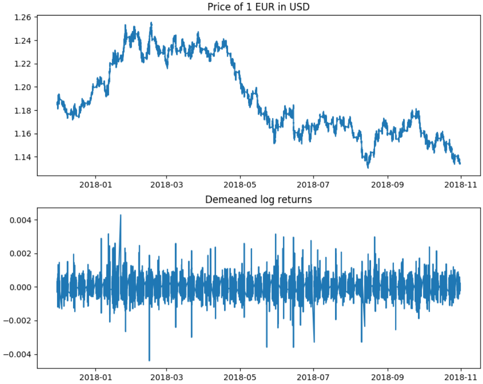

Figure 6: EUR-USD exchange rate data at the minute resolution from November 2017 to October 2018.

SGLD on Exchange Rate Data

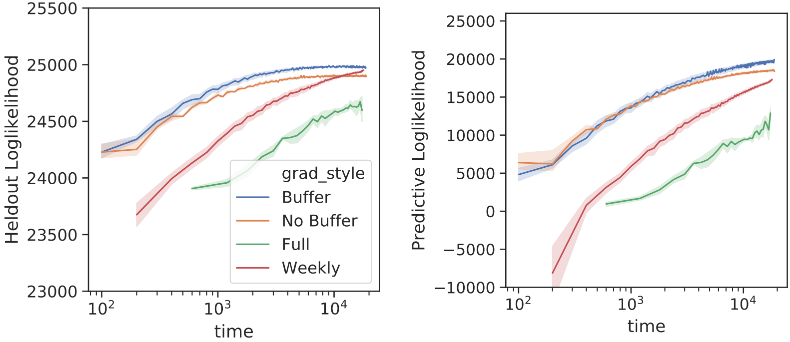

Figure 7: SGLD plots on exchange rate data. (left) heldout-loglikelihood, (right) 3-step ahead predictive loglikelihood.

Buffered Stochastic Gradient Estimate Error

Three types of stochastic gradient error:

- number of particles, N,

Buffered Stochastic Gradient Estimate Error

Three types of stochastic gradient error:

- number of particles, N,
- subsequence length, S,

Buffered Stochastic Gradient Estimate Error

Three types of stochastic gradient error:

- number of particles, N,
- subsequence length, S,
- buffer length, B.

Buffered Stochastic Gradient Estimate Error

Three types of stochastic gradient error:

- number of particles, N,
- subsequence length, S,
- buffer length, B.

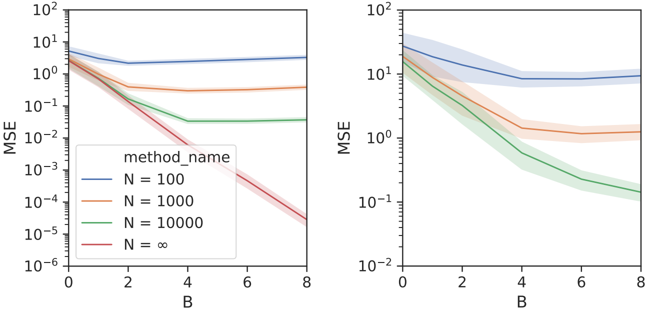

Figure 8: Buffered Stochastic Gradient Estimate Error Plots. (left) LGSSM ϕ, (right) SVM ϕ

Summary

- Developed a particle buffer stochastic gradient estimator for nonlinear SSMs.

Summary

- Developed a particle buffer stochastic gradient estimator for nonlinear SSMs.
- Combined existing literature on buffered SGMCMC (Aicher et al., 2018) with particle filtering for nonlinear SSMs.

Summary

- Developed a particle buffer stochastic gradient estimator for nonlinear SSMs.
- Combined existing literature on buffered SGMCMC (Aicher et al., 2018) with particle filtering for nonlinear SSMs.
- Conducted a theoretical and numerical analysis of the error of our proposed gradient estimator.

Summary

- Developed a particle buffer stochastic gradient estimator for nonlinear SSMs.
- Combined existing literature on buffered SGMCMC (Aicher et al., 2018) with particle filtering for nonlinear SSMs.
- Conducted a theoretical and numerical analysis of the error of our proposed gradient estimator.
- Evaluated our proposed stochastic gradient estimator with SGLD on various models for synthetic and EUR-US exchange rate data.

References

Aicher, C., Ma, Y.-A., Foti, N. J., and Fox, E. B. (2018). Stochastic Gradient MCMC for State Space Models. *arXiv preprint arXiv:1810.09098*.

Aicher, C., Putcha, S., Nemeth, C., Fearnhead, P., and Fox, E. B. (2019). Stochastic gradient MCMC for Nonlinear State Space Models. *arXiv preprint arXiv:1901.10568*.

Cappé, O., Moulines, E., and Rydén, T. (2005). *Inference in Hidden Markov Models*. Springer.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F - Radar and Signal Processing*, 140(2):107–113.

Nemeth, C., Fearnhead, P., and Mihaylova, L. (2016). Particle approximations of the score and observed information matrix for parameter estimation in state–space models with linear computational cost. *Journal of Computational and Graphical Statistics*, 25(4):1138–1157.

Olsson, J., Westerborn, J., et al. (2017). Efficient particle-based online smoothing in general hidden Markov models: the PaRIS algorithm. *Bernoulli*, 23(3):1951–1996.

Poyiadjis, G., Doucet, A., and Singh, S. S. (2011). Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika*, 98(1):65–80.

Welling, M. and Teh, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 681–688.

Any Questions?