Maximum Likelihood Parameter Estimation in State-Space Models

Arnaud Doucet
Department of Statistics, Oxford University

University College London

4th October 2012

State-Space Models

- Let $\{X_t\}_{t\geq 1}$ be a latent/hidden $\mathcal{X}$-valued Markov process with

$$X_1 \sim \mu(\cdot) \text{ and } X_t | (X_{t-1} = x) \sim f(\cdot | x).$$

- Let $\{Y_t\}_{t\geq 1}$ be an $\mathcal{Y}$-valued Markov observation process such that

$$Y_t | (X_t = x) \sim g(\cdot | x).$$

- Particle filters estimate $\{p(x_{1:t} | y_{1:t})\}_{t\geq 1}$ on-line but only estimates of $\{p(x_t | y_{1:t})\}_{t\geq 1}$ and $\{p(y_{1:t})\}_{t\geq 1}$ are reliable.
- Particle smoothing methods allow us to obtain reliable estimates of $\{p(x_t | y_{1:T})\}_{t=1}^{T}$.

State-Space Models with Unknown Parameters

- In most scenarios of interest, the state-space model contains an unknown static parameter $\theta \in \Theta$ so that

$$X_1 \sim \mu_\theta(\cdot) \text{ and } X_t | (X_{t-1} = x) \sim f_\theta(\cdot | x_{t-1}).$$

- The observations $\{Y_t\}_{t \geq 1}$ are conditionally independent given $\{X_t\}_{t \geq 1}$ and θ

$$Y_t | (X_t = x_t) \sim g_\theta(\cdot | x).$$

- **Aim**: We would like to infer θ either on-line or off-line.

Examples

- **Stochastic Volatility model**

$$\begin{aligned} X_t &= \phi X_{t-1} + \sigma V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1) \\ Y_t &= \beta \exp(X_t/2)\, W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1) \end{aligned}$$

where $\theta = (\phi, \sigma^2, \beta)$.

- **Biochemical Network model**

$$\begin{aligned} \Pr\left(X^1_{t+dt}{=}x^1_t{+}1, X^2_{t+dt}{=}x^2_t \middle| x^1_t, x^2_t\right) &= \alpha\, x^1_t\, dt + o(dt), \\ \Pr\left(X^1_{t+dt}{=}x^1_t{-}1, X^2_{t+dt}{=}x^2_t{+}1 \middle| x^1_t, x^2_t\right) &= \beta\, x^1_t\, x^2_t\, dt + o(dt), \\ \Pr\left(X^1_{t+dt}{=}x^1_t, X^2_{t+dt}{=}x^2_t{-}1 \middle| x^1_t, x^2_t\right) &= \gamma\, x^2_t\, dt + o(dt), \end{aligned}$$

with

$$Y_k = X^1_{k\Delta T} + W_k \text{ with } W_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$$

where $\theta = (\alpha, \beta, \gamma)$.

Parameter Inference in State-Space Models

- Online Bayesian parameter inference.
- Offline Maximum Likelihood parameter inference.
- Online Maximum Likelihood parameter inference.

Bayesian Parameter Inference in State-Space Models

- Set a prior $p(\theta)$ on θ so inference relies now on

$$p(\theta, x_{1:t} | y_{1:t}) = \frac{p(\theta, x_{1:t}, y_{1:t})}{p(y_{1:t})}$$

where

$$p(\theta, x_{1:t}, y_{1:t}) = p(\theta) p_\theta(x_{1:t}, y_{1:t})$$

with

$$p_\theta(x_{1:t}, y_{1:t}) = \mu_\theta(x_1) \prod_{k=2}^{t} f_\theta(x_k | x_{k-1}) \prod_{k=1}^{t} g_\theta(y_k | x_k)$$

- We have

$$p(\theta, x_{1:t} | y_{1:t}) = p(\theta | y_{1:t}) p_\theta(x_{1:t} | y_{1:t})$$

- Standard and more sophisticated particle methods to sample from $\{p(\theta, x_{1:t} | y_{1:t})\}_{t \geq 1}$ are ALL unreliabe.

Online Bayesian Parameter Inference

At time $t = 1$

- $\left(\overline{\theta}_1^{(i)}, \overline{X}_1^{(i)}\right) \sim p(\theta)\, \mu_\theta(x_1)$ then
$$\overline{p}(\theta, x_1 | y_1) = \sum_{i=1}^{N} W_1^{(i)} \delta_{\left(\overline{\theta}_1^{(i)}, \overline{X}_1^{(i)}\right)}(\theta, x_1), \quad W_1^{(i)} \propto g_{\overline{\theta}_1^{(i)}}\left(y_1 | \overline{X}_1^{(i)}\right).$$

- $\left(\theta_1^{(i)}, X_1^{(i)}\right) \sim \overline{p}(\theta, x_1 | y_1)$
and $\widehat{p}(\theta, x_1 | y_1) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\theta^{(i)}, X_1^{(i)}}(\theta, x_1)$.

At time $t \geq 2$

- Set $\overline{\theta}_t^{(i)} = \theta_{t-1}^{(i)}$, $\overline{X}_t^{(i)} \sim f_{\overline{\theta}_t^{(i)}}\left(x_t | X_{t-1}^{(i)}\right)$ and $\overline{X}_{1:t}^{(i)} = \left(X_{1:t-1}^{(i)}, \overline{X}_t^{(i)}\right)$
$$\overline{p}(\theta, x_{1:t} | y_1) = \sum_{i=1}^{N} W_t^{(i)} \delta_{\left(\overline{\theta}_t^{(i)}, \overline{X}_{1:t}^{(i)}\right)}(\theta, x_{1:t}), \quad W_t^{(i)} \propto g_{\overline{\theta}_t^{(i)}}\left(y_t | \overline{X}_t^{(i)}\right).$$

- $\left(\theta_t^{(i)}, X_{1:t}^{(i)}\right) \sim \overline{p}(\theta, x_{1:t} | y_{1:t})$ and
$\widehat{p}(\theta, x_{1:t} | y_{1:t}) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\theta_t^{(i)}, X_{1:t}^{(i)}}(\theta, x_{1:t})$.

Online Bayesian Parameter Inference

- Provide consistent estimates but remarkably inefficient (Chopin, 2002). Particles $\left\{\theta_1^{(i)}\right\}$ in Θ space only sampled at time 1: degeneracy problem!
- Consider the extended state $Z_t = (X_t, \theta_t)$ then

$$\begin{aligned} \nu(z_1) &= p(\theta_1)\,\mu_{\theta_1}(x_1), \\ f(z_t|z_{t-1}) &= \delta_{\theta_{t-1}}(\theta_t)\, f_{\theta_t}(x_t|x_{t-1}),\ g(y_t|z_t) = g_{\theta_t}(y_t|x_t); \end{aligned}$$

  i.e. $\theta_t = \theta_1$ for any t with θ_1 from the prior. Exponential stability assumption on $\{p(z_t|y_{1:t})\}_{t\geq 1}$ cannot be satisfied.
- Use MCMC steps on θ so as to jitter $\left\{\theta_t^{(i)}\right\}$; e.g. Andrieu, De Freitas & D. (1999); Fearnhead (2002); Gilks & Berzuini (2001); Carvalho et al. (2010).
- When $p(\theta|y_{1:t}, x_{1:t}) = p(\theta|s_t(x_{1:t}, y_{1:t}))$ where $s_t(x_{1:t}, y_{1:t})$ is a fixed-dimensional vector, "elegant" but still implicitly relies on $\widehat{p}(x_{1:t}|y_{1:t})$ so degeneracy will creep in.

Online Bayesian Parameter Inference

- At time $t-1$, we have

$$\widehat{p}\left(\theta, x_{1:t-1} \middle| y_{1:t-1}\right) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\left(\theta_{t-1}^{(i)}, X_{1:t-1}^{(i)}\right)}\left(\theta, x_{1:t-1}\right),$$

- Set $\overline{\theta}_t^{(i)} = \theta_{t-1}^{(i)}$, sample $\overline{X}_t^{(i)} \sim f_{\overline{\theta}_t^{(i)}}\left(\cdot \middle| X_{t-1}^{(i)}\right)$, set $\overline{X}_{1:t}^{(i)} = \left(X_{1:t-1}^{(i)}, \overline{X}_t^{(i)}\right)$ and

$$\overline{p}\left(\theta, x_{1:t} \middle| y_{1:t}\right) = \sum_{i=1}^{N} W_t^{(i)} \delta_{\left(\overline{\theta}_t^{(i)}, \overline{X}_{1:t}^{(i)}\right)}\left(\theta, x_{1:t}\right),$$

$$W_t^{(i)} \propto g_{\overline{\theta}_t^{(i)}}\left(y_t \middle| \overline{X}_t^{(i)}\right).$$

- Resample $\left(\theta_t'^{(i)}, X_{1:t}^{(i)}\right) \sim \overline{p}\left(\theta, x_{1:t} \middle| y_{1:t}\right)$ then sample $\theta_t^{(i)} \sim p\left(\theta \middle| y_{1:t}, X_{1:t}^{(i)}\right)$ to obtain $\widehat{p}\left(\theta, x_{1:t} \middle| y_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\left(\theta_t^{(i)}, X_{1:t}^{(i)}\right)}\left(\theta, x_{1:t}\right)$.

A Toy Example

- Linear Gaussian state-space model

$$X_t = \theta X_{t-1} + \sigma_V V_t, \quad V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = X_t + \sigma_W W_t, \quad W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

- We set $p(\theta) \propto 1_{(-1,1)}(\theta)$ so

$$p(\theta | y_{1:t}, x_{1:t}) \propto \mathcal{N}\left(\theta; m_t, \sigma_t^2\right) 1_{(-1,1)}(\theta)$$

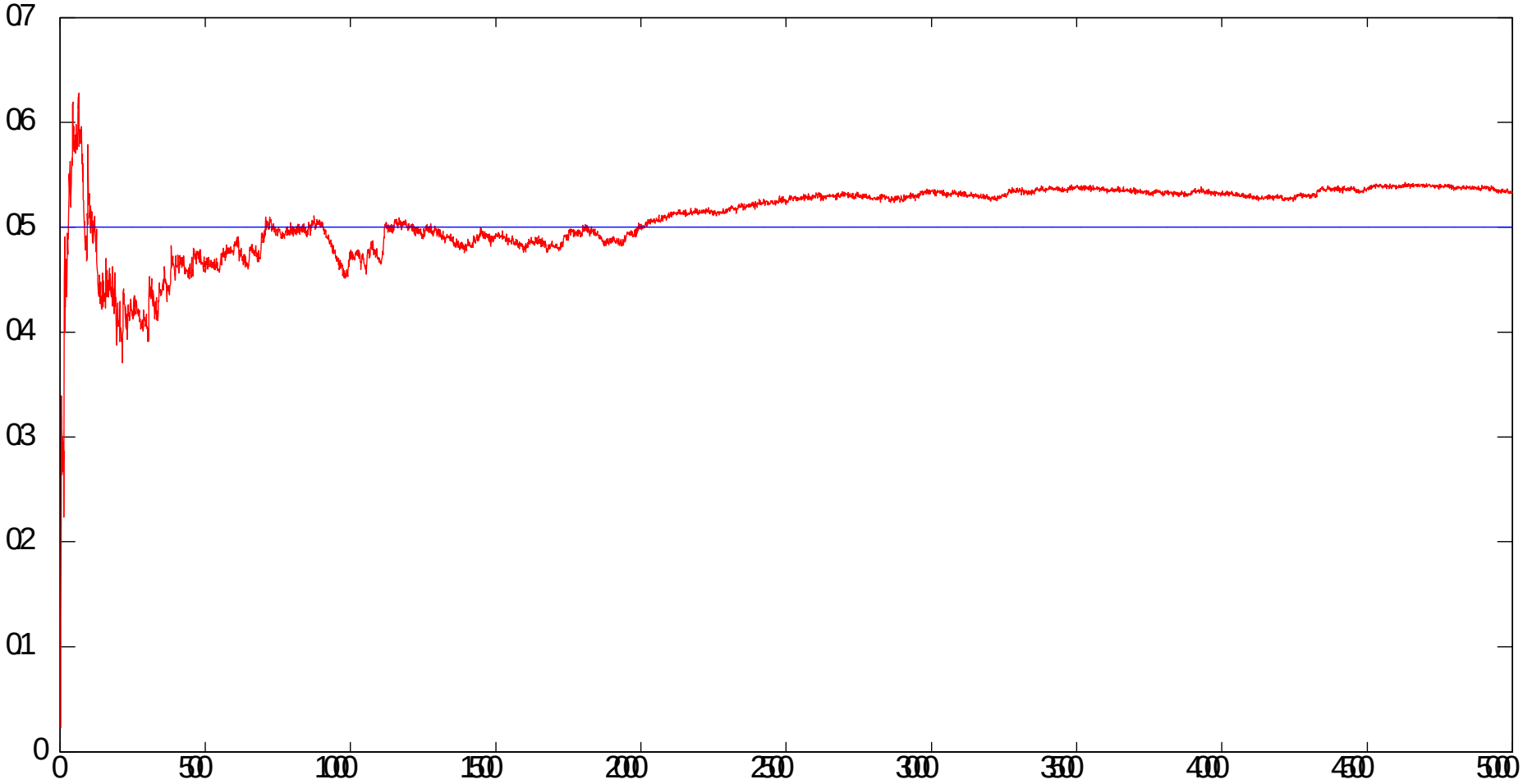
where

$$\sigma_t^2 = S_{2,t}^{-1}, \; m_t = S_{2,t}^{-1} S_{1,t}$$

with

$$S_{1,t} = \sum_{k=2}^{t} x_{k-1} x_k, \quad S_{2,t} = \sum_{k=2}^{t} x_{k-1}^2$$

Illustration of the Degeneracy Problem



SMC estimate of $\mathbb{E}\left[\theta \middle| y_{1:t}\right]$, as t increases the degeneracy creeps in.

Another Toy Example

- Linear Gaussian state-space model

$$X_t = \rho X_{t-1} + V_t, \;\; V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = X_t + \sigma W_t, \;\; W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

- We set $\rho \sim \mathcal{U}_{(-1,1)}$ and $\sigma^2 \sim \mathcal{IG}(1,1)$.
- We use particle filter with perfect adaptation and Gibbs moves with $N = 10000$; particle learning (Andrieu, D. & De Freitas, 1999; Carvalho et al., 2010)
- 50 runs of the particle method vs ground truth obtained using Kalman filter on states and grid on parameters.

Another Illustration of Degeneracy for Particle Learning

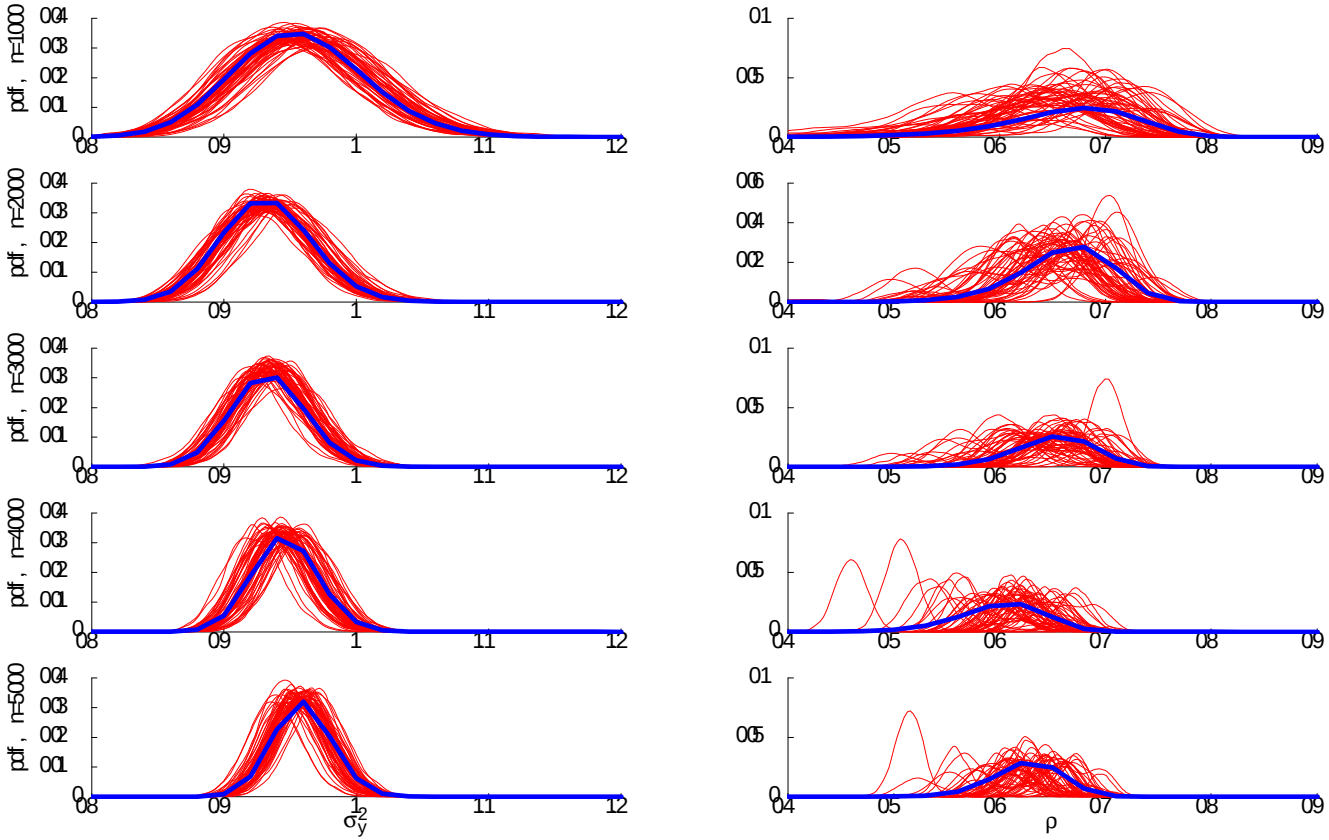


Figure: Estimates of $p(\rho | y_{1:t})$ and $p(\sigma^2 | y_{1:t})$ over 50 runs (red) vs ground truth (blue) for $t = 10^3, 2.10^3, ..., 5.10^3$ for $N = 10^4$ (Kantas et al., 2012)

Stepping Back

- For fixed θ, $\mathbb{V}\left[\widehat{p}_\theta\left(y_{1:t}\right)/p_\theta\left(y_{1:t}\right)\right]$ is in $\mathcal{O}\left(t/N\right)$.
- In a Bayesian context, $p\left(\theta|\, y_{1:t}\right) \propto p_\theta\left(y_{1:t}\right) p\left(\theta\right)$ so we implicitly need to compute $p_\theta\left(y_{1:t}\right)$ at each particle location $\theta^{(i)}$.
- It appears impossible to obtain uniformly in time stable estimates of $\left\{p\left(\theta|\, y_{1:t}\right)\right\}_{t\geq 1}$ for a fixed N.
- However for a given time horizon T, we can use PF to sample efficently from $p\left(\theta|\, y_{1:T}\right)$; see Lecture 3.

Likelihood Function Estimation

- Let $y_{1:T}$ being given, the log-(marginal) likelihood is given by

$$\ell(\theta) = \log p_\theta \left(y_{1:T} \right).$$

- For any $\theta \in \Theta$, one can estimate $\ell(\theta)$ using particle methods, variance $\mathcal{O}\left(T/N\right)$.
- Direct maximization of $\ell(\theta)$ difficult as estimate $\widehat{\ell}(\theta)$ is not a smooth function of θ even for fixed random seed.
- For $\dim\left(X_t\right) = 1$, we can obtain smooth estimate of log-likelihood function by using a smoothed resampling step (e.g. Pitt, 2011); i.e. piecewise linear approximation of $\Pr\left(X_t < x \middle| y_{1:t}\right)$.
- For $\dim\left(X_t\right) > 1$, we can obtain estimates of $\ell(\theta)$ highly positively correlated for neigbouring values in Θ (e.g. Lee, 2008).

Gradient Ascent

- To maximise $\ell(\theta)$ w.r.t θ, use at iteration $k+1$

$$\theta_{k+1} = \theta_k + \gamma_k \left. \nabla \ell(\theta) \right|_{\theta=\theta_k}$$

where $\left. \nabla \ell(\theta) \right|_{\theta=\theta_k}$ is the so-called score vector.

- $\left. \nabla \ell(\theta) \right|_{\theta=\theta_k}$ can be estimated using finite differences but more efficiently using Fisher's identity

$$\nabla \ell(\theta) = \int \nabla \log p_\theta \left(x_{1:T}, y_{1:T} \right) \; p_\theta \left(x_{1:T} \middle| y_{1:T} \right) dx_{1:T}$$

where

$$\begin{aligned} &\nabla \log p_\theta \left(x_{1:T}, y_{1:T} \right) = \nabla \log \mu_\theta \left(x_1 \right) \\ &+ \textstyle\sum_{t=2}^{T} \nabla \log f_\theta \left(x_t \middle| x_{t-1} \right) + \sum_{t=1}^{T} \nabla \log g_\theta \left(y_t \middle| x_t \right). \end{aligned}$$

Particle Calculation of the Score Vector

- We have

$$\nabla \ell(\theta) = \int \{\nabla \log \mu_\theta(x_1) + \nabla \log g_\theta(y_1 | x_1)\} p_\theta(x_1 | y_{1:T}) \, dx_1$$

$$+ \sum_{t=2}^{T} \int \{\nabla \log f_\theta(x_t | x_{t-1}) + \nabla \log g_\theta(y_t | x_t)\} p_\theta(x_{t-1}, x_t | y_{1:T}) \, dx_{t-1} dx_t$$

- To approximate $\nabla \ell(\theta)$, we just need particle approximations of $\{p_\theta(x_{t-1}, x_t | y_{1:T})\}_{t=2}^{T}$.
- All the particle smoothing methods detailed before can be applied.
- Similar "smoothed additive functionals" have to be computed when implementing the Expectation-Maximization.

Comparison Direct Method vs FB

- We want to estimate
$\overline{\varphi}_T = \sum_{t=1}^{T} \int \varphi(x_{t-1}, x_t, y_t)\, p(x_{t-1}, x_t \,|\, y_{1:T})\, dx_{t-1} dx_t$.

Method	Direct	FB
# particles	N	N
cost	$\mathcal{O}(TN)$	$\mathcal{O}(TN^2), \mathcal{O}(TN)$
Var.	$\mathcal{O}(T^2/N)$	$\mathcal{O}(T/N)$
Bias	$\mathcal{O}(T/N)$	$\mathcal{O}(T/N)$
MSE=Bias^2+Var	$\mathcal{O}(T^2/N)$	$\mathcal{O}(T^2/N^2)$

- "Fast" implementations FB of computational complexity $\mathcal{O}(NT)$ outperform direct approach as MSE is $\mathcal{O}(T^2/N^2)$ whereas it is $\mathcal{O}(T^2/N)$ for direct SMC.
- "Naive" implementations FB and TF have MSE of same order as direct method for fixed computational complexity but MSE is bias dominated for FB/TF whereas it is variance dominated for Direct SMC.

Experimental Results

- Consider a linear Gaussian model

$$X_t = \phi X_{t-1} + \sigma_v V_t, \;\; V_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$
$$Y_t = cX_t + \sigma_w W_t, \;\; W_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

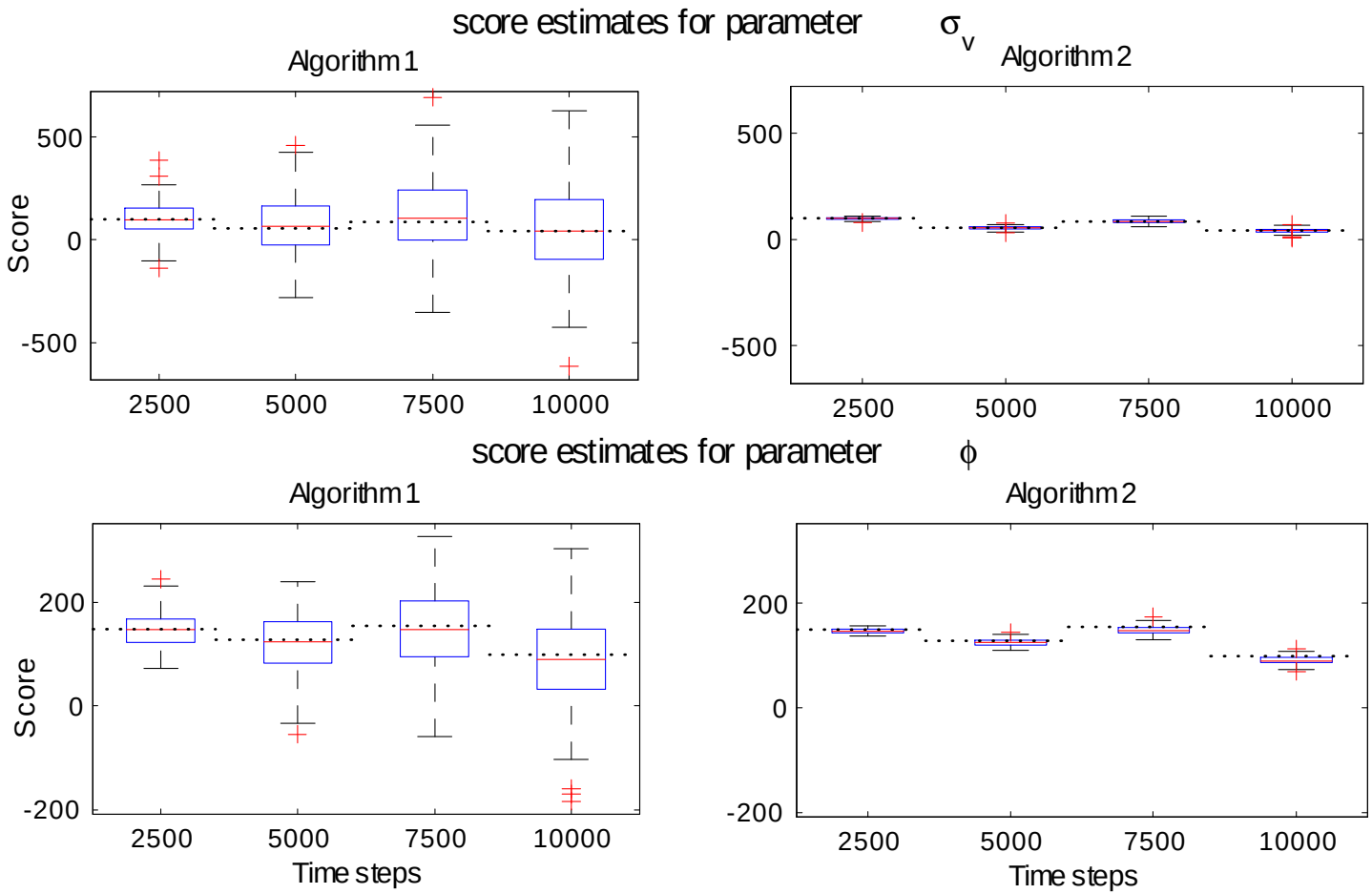
- We simulate 10,000 observations and compute particle estimates of

$$\int \varphi_T(x_{1:T}) \;\; p(x_{1:T} | y_{1:T}) \, dx_{1:T}$$

for 4 different additive functionals
$\varphi_t(x_{1:t}) = \varphi_{t-1}(x_{1:t-1}) + \varphi(x_{t-1}, x_t, y_t)$ including
$\varphi^1(x_{t-1}, x_t, y_t) = x_{t-1}x_t$, $\varphi^2(x_{t-1}, x_t, y_t) = x_t^2$. [Ground truth can be computed using Kalman smoother.]

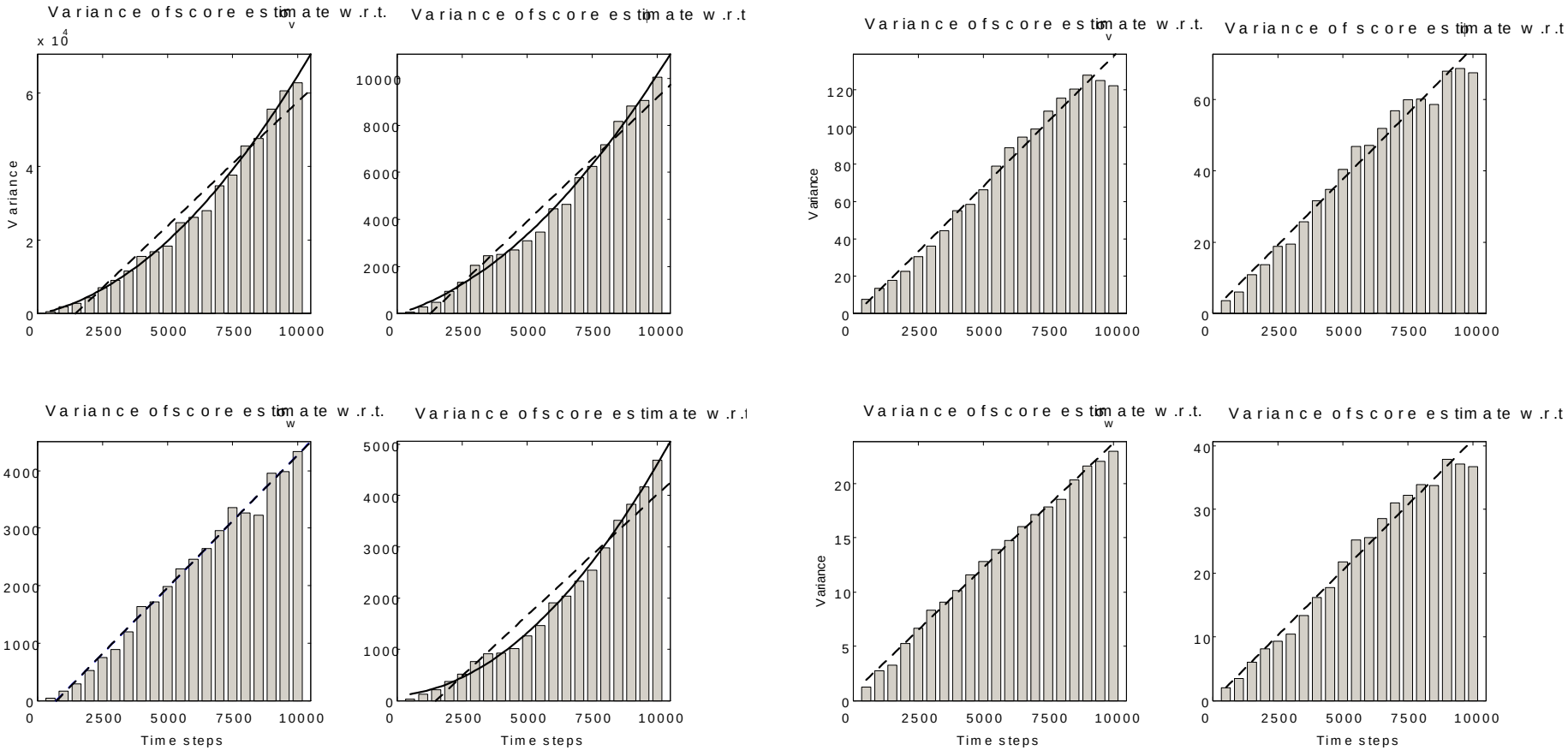
- We use 100 replications on the same dataset to estimate the empirical variance.

Boxplots of Direct vs FB Estimates



Direct (left) vs FB (right)

Empirical Variance for Direct vs FB Estimates



Direct (left) vs FB (right); the vertical scale is different

Online ML Parameter Inference

- *Recursive maximum likelihood* (Titterington, 1984; LeGland & Mevel, 1997) proceeds as follows

$$\theta_{t+1} = \theta_t + \gamma_t \, \nabla \log \, p_{\theta_{1:t}} \left(y_t \middle| \, y_{1:t-1} \right)$$

where $p_{\theta_{1:t}} \left(y_t \middle| \, y_{1:t-1} \right)$ is computed using θ_k at time k and $\sum_t \gamma_t = \infty$, $\sum_t \gamma_t^2 < \infty$. Under regularity conditions, this converges towards a local maximum of the (average) log-likelihood.

- Note that

$$\nabla \log \, p_{\theta_{1:t}} \left(y_t \middle| \, y_{1:t-1} \right) = \nabla \log \, p_{\theta_{1:t}} \left(y_{1:t} \right) - \nabla \log \, p_{\theta_{1:t-1}} \left(y_{1:t-1} \right)$$

is given by the difference of two pseudo-score vectors where

$$\nabla \log \, p_{\theta_{1:t}} \left(y_{1:t} \right) := \int \left(\sum_{k=2}^{t} \nabla \log f_\theta \left(x_k \middle| \, x_{k-1} \right) \big|_{\theta_k} + \nabla \log g_\theta \left(y_k \middle| \, x_k \right) \big|_{\theta_k} \right) p_{\theta_{1:t}} \left(x_{1:t} \middle| \, y_{1:t} \right) dx_{1:t}.$$

Online ML Particle Parameter Inference

- Particle approximation follows

$$\theta_{t+1} = \theta_t + \gamma_t \, \widehat{\nabla \log} \, p_{\theta_{1:t}} \left(y_t \middle| y_{1:t-1} \right)$$

where

$$\widehat{\nabla \log} \, p_{\theta_{1:t}} \left(y_t \middle| y_{1:t-1} \right) = \widehat{\nabla \log} \, p_{\theta_{1:t}} \left(y_{1:t} \right) - \widehat{\nabla \log} \, p_{\theta_{1:t-1}} \left(y_{1:t-1} \right)$$

is given by the difference of particle estimates of pseudo-score vectors (Poyadjis, D. & Singh, 2011).

- Asymptotic variance of $\widehat{\nabla \log} \, p_{\theta_{1:t}} \left(y_t \middle| y_{1:t-1} \right)$ is uniformly bounded in $\mathcal{O}\left(1/N\right)$ for FB estimate whereas it is $\mathcal{O}\left(t/N\right)$ for direct particle method (Del Moral, D. & Singh, 2011). Bias is $\mathcal{O}\left(1/N\right)$ in both cases.
- **Major Problem**: If we use FB, this is not an online algorithm anymore as it requires a backward pass of order $\mathcal{O}\left(t\right)$ to approximate $\nabla \log \, p_{\theta_{1:t}} \left(y_{1:t} \right)$ ...

Variance of the Gradient Estimate for Direct vs FB

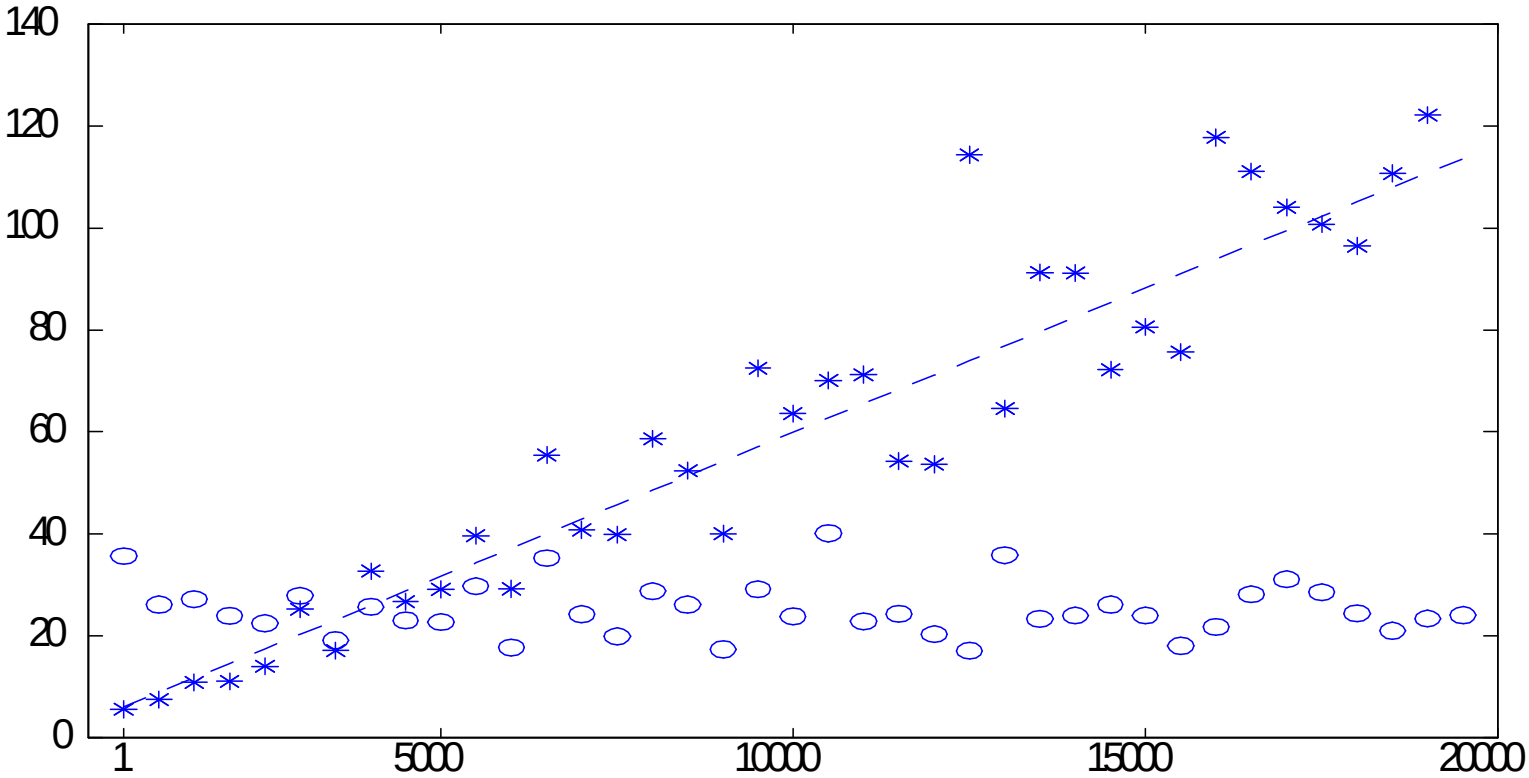


Figure: Empirical variance of the gradient estimate for standard versus FB approximations (SV model)

Online Particle ML Inference using Direct Approach

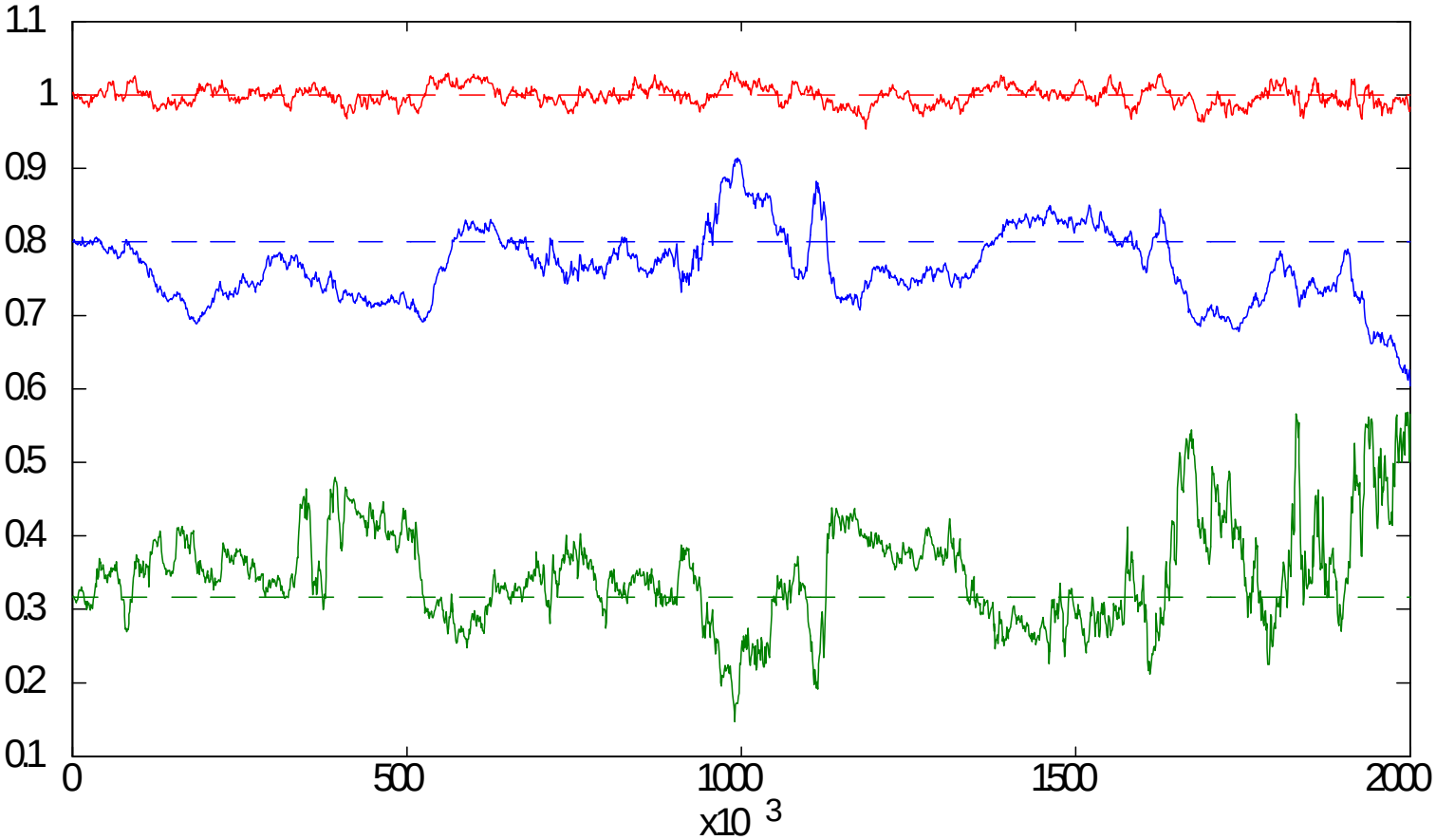


Figure: $N = 10,000$ particles, online parameter estimates for SV model.

Online Particle ML Inference using FB

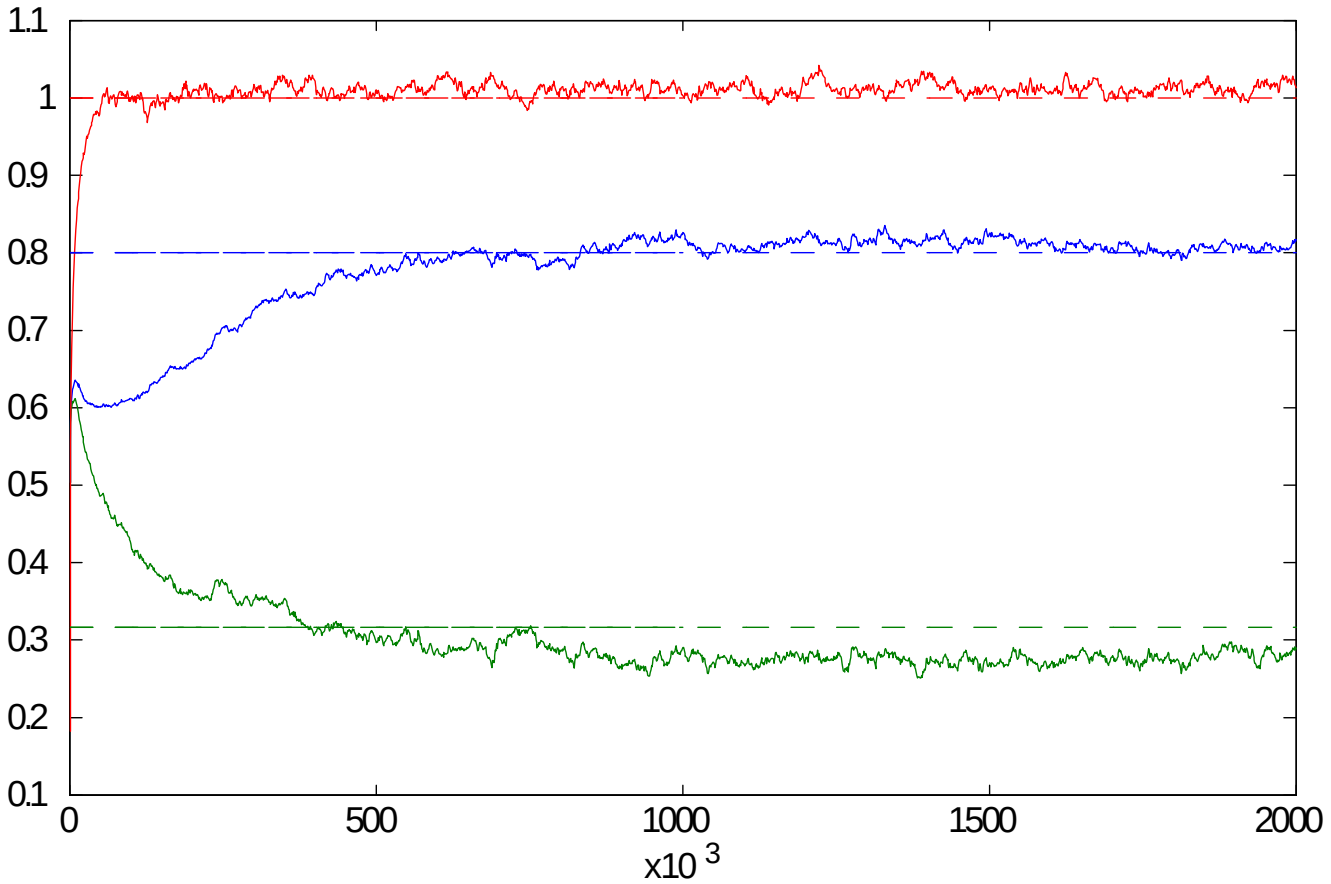


Figure: $N = 50$ particles, online parameter estimates for SV model.

Forward only Smoothing

- Dynamic programming allows us to compute in a single forward pass the FB estimates of

$$\varphi_t^\theta = \int_t \varphi_t\left(x_{1:t}\right)\ p_\theta\left(x_{1:t}\middle|\, y_{1:t}\right) dx_{1:t}$$

where

$$\varphi_t\left(x_{1:t}\right) = \sum_{k=1}^{t} \varphi\left(x_{k-1}, x_k, y_k\right)$$

- Forward Backward (FB) decomposition states

$$p_\theta\left(x_{1:T}\middle|\, y_{1:T}\right) = p_\theta\left(x_T\middle|\, y_{1:T}\right) \prod_{t=1}^{T-1} p_\theta\left(x_t\middle|\, y_{1:t}, x_{t+1}\right)$$

where $p_\theta\left(x_t\middle|\, y_{1:t}, x_{t+1}\right) = \frac{f_\theta\left(x_{t+1}|x_t\right) p_\theta\left(x_t|y_{1:t}\right)}{p_\theta\left(x_{t+1}|y_{1:t}\right)}$.

- Conditioned upon $y_{1:T}$, $\{X_t\}_{t=1}^T$ is a backward Markov chain of initial distribution $p\left(x_T\middle|\, y_{1:T}\right)$ and inhomogeneous Markov transitions $\left\{p_\theta\left(x_t\middle|\, y_{1:t}, x_{t+1}\right)\right\}_{t=1}^{T-1}$ independent of T.

Forward only Smoothing

- We have

$$\varphi_t^\theta = \int \varphi_t(x_{1:t})\; p_\theta(x_{1:t-1} | y_{1:t-1}, x_t)\, dx_{1:t-1}$$

$$= \int \left\{ \underbrace{\int \varphi_t(x_{1:t})\; p_\theta(x_{1:t-1} | y_{1:t-1}, x_t)\, dx_{1:t-1}}_{V_t^\theta(x_t)} \right\} p_\theta(x_t | y_{1:t})\, dx_t$$

- *Forward smoothing recursion*

$$V_t^\theta(x_t) = \int \left[V_{t-1}^\theta(x_{t-1}) + \varphi(x_{t-1:t}, y_t) \right] p_\theta(x_{t-1} | y_{1:t-1}, x_t)\, dx_{t-1}$$

- Appears implicitly in Elliott, Aggoun & Moore (1996), Ford (1998) and rediscovered a few times... Presentation follows here (Del Moral, D. & Singh, 2009).

Forward only Smoothing

- *Forward smoothing recursion*

$$V_t^\theta(x_t) = \int \left[V_{t-1}^\theta(x_{t-1}) + \varphi(x_{t-1:t}, y_t) \right] p_\theta(x_{t-1} | y_{1:t-1}, x_t) \, dx_{t-1}$$

- Proof is trivial

$$\begin{aligned} V_t^\theta(x_t) &= \int \varphi_t(x_{1:t}) \; p_\theta(x_{1:t-1} | y_{1:t-1}, x_t) \, dx_{1:t-1} \\ &= \int \left[\varphi_{t-1}(x_{1:t-1}) + \varphi(x_{t-1:t}, y_t) \right] \; p_\theta(x_{1:t-2} | y_{1:t-2}, x_{t-1}) \\ &\qquad \times p_\theta(x_{t-1} | y_{1:t-1}, x_t) \, dx_{1:t-1} \\ &= \int \{ \underbrace{\int \varphi_{t-1}(x_{1:t-1}) \, p_\theta(x_{1:t-2} | y_{1:t-2}, x_{t-1}) \, dx_{1:t-2}}_{V_{t-1}^\theta(x_{t-1})} \\ &\qquad + \varphi(x_{t-1:t}, y_t) \} \; p_\theta(x_{t-1} | y_{1:t-1}, x_t) \, dx_{t-1} \end{aligned}$$

- Exact implementation possible for finite state-space and linear Gaussian models.

Particle Forward only Smoothing

- At time $t-1$, we have $\widehat{p}_\theta\left(x_{t-1}|\, y_{1:t-1}\right) = \frac{1}{N}\sum_{i=1}^{N}\delta_{X_{t-1}^{(i)}}\left(x_{t-1}\right)$ and $\left\{\widehat{V}_{t-1}^{\theta}\left(X_{t-1}^{(i)}\right)\right\}_{1\leq i\leq N}$.
- At time t, compute $\widehat{p}_\theta\left(x_t|\, y_{1:t}\right) = \sum_{i=1}^{N} W_t^{(i)}\delta_{X_t^{(i)}}\left(x_t\right)$ and set

$$\widehat{V}_t^{\theta}\left(X_t^{(i)}\right) = \int\left\{\widehat{V}_{t-1}^{\theta}\left(x_{t-1}\right)+\varphi\left(x_{t-1},x_t,y_t\right)\right\}\widehat{p}_\theta\left(x_{t-1}|\, y_{1:t-1},X_t^{(i)}\right)dx_{t-1}$$
$$= \frac{\sum_{j=1}^{N} f_\theta\left(X_t^{(i)}|X_{t-1}^{(j)}\right)\left[\widehat{V}_{t-1}^{\theta}\left(X_{t-1}^{(j)}\right)+\varphi\left(X_{t-1}^{(j)},X_t^{(i)},y_t\right)\right]}{\sum_{j=1}^{N} f_\theta\left(X_t^{(i)}|X_{t-1}^{(j)}\right)},$$

$$\widehat{\varphi}_t^{\theta} = \frac{1}{N}\sum_{i=1}^{N}\widehat{V}_t^{\theta}\left(X_t^{(i)}\right).$$

- This estimate is exactly the same as the Particle FB estimate, computational complexity $\mathcal{O}\left(N^2\right)$.

Online Particle ML Inference

- At time $t-1$, we have $\widehat{p}_{\theta_{1:t-1}}(x_{t-1}|y_{1:t-1})$, $\left\{\widehat{V}_{t-1}^{\theta_{1:t-1}}\left(X_{t-1}^{(i)}\right)\right\}$, $\widehat{\nabla \log}\, p_{\theta_{1:t-1}}(y_{1:t-1}) = \int \widehat{V}_{t-1}^{\theta_{1:t-1}}(x_{t-1})\, \widehat{p}_{\theta_{1:t-1}}(x_{t-1}|y_{1:t-1})\, dx_{t-1}$ and get θ_t.
- At time t, use your favourite PF to compute $\widehat{p}_{\theta_{1:t}}(x_t|y_{1:t})$ and

$$\widehat{V}_t^{\theta_{1:t}}\left(X_t^{(i)}\right) = \int \left\{\widehat{V}_{t-1}^{\theta_{1:t-1}}(x_{t-1}) + \varphi(x_{t-1}, x_t, y_t)\right\} \times \widehat{p}_{\theta_{1:t}}\left(x_{t-1}|y_{1:t-1}, X_t^{(i)}\right) dx_{t-1},$$

$$\varphi(x_{t-1:t}, y_t) = \nabla \log f_\theta(x_t|x_{t-1})|_{\theta_t} + \nabla \log g_\theta(y_t|x_t)|_{\theta_t}$$

and

$$\widehat{\nabla \log}\, p_{\theta_{1:t}}(y_{1:t}) = \int \widehat{V}_t^{\theta_{1:t}}(x_t)\, \widehat{p}_{\theta_{1:t}}(x_t|y_{1:t})\, dx_t$$

- Parameter update

$$\theta_{t+1} = \theta_t + \gamma_t \left(\widehat{\nabla \log}\, p_{\theta_{1:t}}(y_{1:t}) - \widehat{\nabla \log}\, p_{\theta_{1:t-1}}(y_{1:t-1})\right)$$

Summary

- Online Bayesian parameter inference using particle methods is yet an unsolved problem.
- Particle smoothing techniques can be used to perform off-line and on-line ML parameter estimation.
- Observed information matrix can also be evaluated online in a stable manner.
- For online inference, computational complexity is $\mathcal{O}\left(N^2\right)$ at each time step and requires evaluating $f_\theta\left(x_t \middle| x_{t-1}\right)$.