Nonlinear Particle Filtering and Learning

Nick Polson
University of Chicago
SAMSI 2008

M. Johannes and S-M Yae

Overview

- Nonlinear Filtering and Learning
- Parameter Learning $\equiv$ Filtering conditional Sufficient Statistics
- Slice variables give conditional sufficient statistics for *any* model
  Carvalho, Johannes, Lopes and Polson (2008) Particle Learning and Smoothing

Nonlinear State Space Model

- CJLP (2008) develop particle learning for a general state space model

$$\text{Observation equation:} \quad y_{t+1} \sim p\left(y_{t+1}|x_{t+1},\theta\right)$$
$$\text{Evolution equation:} \quad x_{t+1} \sim p\left(x_{t+1}|x_t,\theta\right)$$

  with initial state distribution $p\left(x_0|\theta\right)$ and prior $p(\theta)$.
- Sequential Parameter Learning and filtering problem

$$p\left(x_t,\theta|y^t\right)$$

- Relies on conditional sufficient statistics s_t
- Nonlinearity in state evolution straightforward
  Observation and Parameter nonlinearity?

Bayesian Learning

- Bayes rule,

$$p\left(\theta|y^{t+1}\right) \propto p\left(y_{t+1}|\theta\right) p\left(\theta|y^{t}\right).$$

- Bayes without tears:
  Sample $\theta^{(i)} \sim p(\theta)$ and re-sample with $p(y^t|\theta^{(i)})$ is very efficient.
- Sequential importance sampling suffers from the same issue as do MCEM methods
- CJLP propose a re-sample-propagate algorithm which needs conditional sufficient statistics

Updating Slice Sufficient Statistics

- Auxiliary slice variable u_t,

$$p\left(\theta, u_{t+1} | u^t, y^{t+1}\right) \propto p\left(u_{t+1} | y_{t+1}, \theta\right) p\left(\theta | u^t, y^t\right)$$
$$\propto \mathbb{I}\left[u_{t+1} < p\left(y_{t+1} | \theta\right)\right] p\left(\theta | u^t, y^t\right)$$
$$\propto \mathbb{I}_{a(u_{t+1}, y_{t+1}) \leq \theta \leq A(u_{t+1}, y_{t+1})} p\left(\theta | u^t, y^t\right)$$

- Assume that the slice region is invertible:

$$\left[u_{t+1} < p\left(y_{t+1} | \theta\right)\right] \Leftrightarrow \left[a\left(u_{t+1}, y_{t+1}\right) \leq \theta \leq A\left(u_{t+1}, y_{t+1}\right)\right].$$

- Posterior $p\left(\theta | u^t, y^t\right) \propto \mathbb{I}_{[a_t < \theta < A_t]} p\left(\theta\right)$ gives recursion for $p\left(\theta | u^{t+1}, y^{t+1}\right)$ proportional to

$$\mathbb{I}\left[a\left(u_{t+1}, y_{t+1}\right) < \theta < A\left(u_{t+1}, y_{t+1}\right)\right] \mathbb{I}\left[a_t < \theta < A_t\right] p\left(\theta\right).$$

- Combining the indicators implies that

$$\mathbb{I}\left[a\left(u_{t+1}, y_{t+1}\right) < \theta < A\left(u_{t+1}, y_{t+1}\right)\right] \mathbb{I}\left[a_t < \theta < A_t\right] = \mathbb{I}\left[a_{t+1}, A_{t+1}\right]$$

where the fixed-dimension sufficient statistics are defined via the recursion

$$\begin{aligned} a_{t+1} &= \max\left(a_t, a\left(u_{t+1}, y_{t+1}\right)\right) \\ A_{t+1} &= \min\left(A_t, A\left(u_{t+1}, y_{t+1}\right)\right). \end{aligned}$$

Thus, defining $s_t = (a_t, A_t)$, this generates a recursive mapping $s_{t+1} = \mathcal{S}\left(a_t, A_t, u_{t+1}, y_{t+1}\right)$, as required.

Posterior: Mixture of Uniforms

- Posterior as the Monte Carlo average

$$
\begin{aligned}
p^N\left(\theta|y^t\right) &= \frac{1}{N}\sum\nolimits_{i=1}^{N} p\left(\theta|\left(a_t, A_t\right)^{(i)}\right) \\
&= \frac{1}{N}\sum\nolimits_{i=1}^{N} \mathbb{I}\left[a_t^{(i)} \leq \theta \leq A_t^{(i)}\right] \\
&\rightarrow E\left[p\left(\theta|a_t, A_t\right)|y^t\right] = p\left(\theta|y^t\right),
\end{aligned}
$$

as N increases. This result is reminiscent of Feller (1943), who represents distributions as mixtures of uniforms.

Slicing the Likelihood

- Likelihood slice region invertible:

  $$0 \leq u_{t+1} \leq p\left(y_{t+1}|x_{t+1},\theta\right) \ \text{ implies } \ a \leq \theta \leq A,$$

  where we update

  $$a = a\left(u_{t+1}, y_{t+1}, x_{t+1}\right) \text{ and } A = A\left(u_{t+1}, y_{t+1}, x_{t+1}\right). \tag{1}$$

- Induction argument for sufficient statistics $s_t = (a_t, A_t, s_t^x)$, where

  $$p\left(\theta|s_t\right) \propto \mathbb{I}_{[a_t \leq \theta \leq A_t]} p\left(\theta|s_t^x\right).$$

- Essentially, a_t and A_t constrain only a subset of the parameter vector. Compute $p\left(\theta|s_{t+1}\right)$ given y_{t+1}, x_{t+1}, and u_{t+1}

$$
\begin{aligned}
p\left(\theta|s_{t+1}\right) &\propto p\left(u_{t+1}, x_{t+1}, y_{t+1}|x_t, \theta\right) p\left(\theta|s_t\right) \\
&\propto p\left(u_{t+1}, y_{t+1}|x_{t+1}, \theta\right) p\left(x_{t+1}|x_t, \theta\right) \mathbb{I}_{[a_t \leq \theta \leq A_t]} p\left(\theta|s_t^x\right) \\
&\propto \mathbb{I}_{[a \leq \theta \leq A]} \mathbb{I}_{[a_t \leq \theta \leq A_t]} p\left(x_{t+1}|x_t, \theta\right) p\left(\theta|s_t^x\right).
\end{aligned}
$$

- The posterior is

$$
p\left(\theta|s_{t+1}\right) = \mathbb{I}_{[a_{t+1} \leq \theta \leq A_{t+1}]} p\left(\theta|s_{t+1}^x\right),
$$

Algorithm

Example

- Step 1: Re-sample $(x_t, s_t, \theta)^{(i)}$ with

$$w^{(i)} = \frac{p\left(y_{t+1} | (x_t, \theta)^{(i)}\right)}{\sum_{j=1}^{N} p\left(y_{t+1} | (x_t, \theta)^{(j)}\right)}$$

- Step 2. Propogate states: $x_{t+1}^{(i)} \sim p(x_{t+1} | (x_t, \theta)^{(i)}, y_{t+1})$.
- Step 3. Draw auxiliary variables:

$$\begin{aligned} u_{t+1}^{(i)} &\sim p(u_{t+1} | \left(x_{t+1}, \theta\right)^{(i)}, y_{t+1}) \\ &\sim \mathcal{U}\left[cp\left(y_{t+1} | \left(x_{t+1}, \theta\right)^{(i)}\right), p\left(y_{t+1} | \left(x_{t+1}, \theta\right)^{(i)}\right)\right] \end{aligned}$$

- Step 4: Update s_{t+1} and draw $\theta^{(i)} \sim p(\theta | s_{t+1}^{(i)})$.

Marginalisation increases efficiency e.g. $p(y_{t+1} | x_t, a_t, A_t)$

Applications: Exponential State Space Model

Example

- Nonlinear exponential state space model

$$
\begin{aligned}
y_t &= e^{-\gamma x_t} + \sigma \varepsilon_t^y \\
x_t &= \alpha + \beta x_{t-1} + \sigma_x \varepsilon_t^x
\end{aligned}
$$

- Normal/inverse Gamma prior for $\left(\alpha, \beta, \sigma_x^2\right) \sim \mathcal{N}\left(c, C\sigma_x^2\right) \mathcal{IG}\left(d, D\right)$.
- Normal/inverse Gamma conjugate priors,

$$
p\left(\alpha, \beta, \sigma_x^2 | s_t^x\right) = \mathcal{N}\left(c_t, C_t \sigma_x^2\right) \mathcal{IG}\left(d_t, D_t\right),
$$

- Marginal likelihood

$$
p\left(y_t | s_t, \gamma, b_{t-1}, B_{t-1}\right) \propto \left(1 + \frac{\left(y_t - e^{-\gamma s_t}\right)^2}{2B_{t-1}}\right)^{-\frac{b_{t-1}+1}{2}}
$$

- Parameter posteriors $p(\gamma|y_t, x_t, s_{t-1})$ given by $\overline{a} \leq \gamma \leq \overline{A}$ where

$$[\overline{a}, \overline{A}] = \left[-\frac{1}{s_t} \ln\left(y_t + K\right), -\frac{1}{s_t} \ln\left(y_t - K\right)^+ \right]$$

where

$$K = \sqrt{2B_{t-1}\left(u_t^{-\frac{1}{b_{t-1}+1}} - 1\right)}$$

Nonlinear Model

Example

- $$y_t = \frac{|x_t|^\alpha}{20} + \sigma \varepsilon_t^y, \quad \text{for} \;\; t = 1, \ldots, 300$$
  $$x_{t+1} = \beta_1 x_t + \frac{\beta_2 x_t}{1 + x_t^2} + \sigma_x \sqrt{\lambda_{t+1}} \varepsilon_{t+1}^x$$
  where $\lambda_{t+1} \sim \mathcal{IG}\left(\frac{\nu}{2}, \frac{\nu}{2}\right)$ and ε_t^y and ε_t^x are independent standard normal.
- Extension of CPS (1992).

Data

- To illustrate the algorithm, we simulate artificial datasets for the following parameters: $\gamma = 1$, $\alpha = 0$, $\beta = 0.9$, $\sigma = 0.3$, and $\sigma_x = 0.2$. The priors parameters are given by

$$\begin{aligned}
&\gamma \sim \mathcal{U}(0, 4) \\
&b = 10, \; B = 0.72 \\
&c = (0, 0.9), \; C = diag(0.1, 0.2) \\
&d = 20, \; D = 0.76.
\end{aligned}$$

- Nonlinear parameter α is given by $a \le \alpha \le A$ where

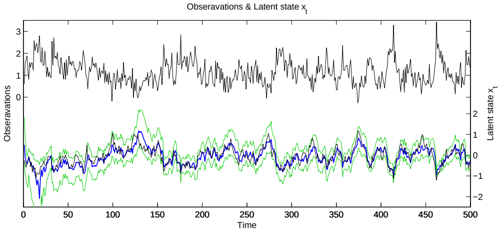
$$[a, A] = \left[\frac{1}{\ln |s_t|} \ln 20 \left(y_t + K \right), \frac{1}{\ln |s_t|} \ln 20 \left(y_t - K \right)^+ \right]$$

$$K = \sqrt{2B_t \left(u_t^{-\frac{1}{b_t+1}} - 1 \right)}$$

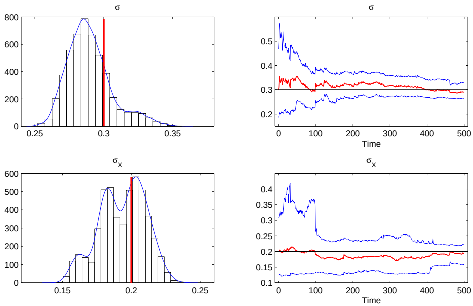
- Data $\alpha = 2, \nu = 3$, $\beta_x = (0.5, 25, 8)$, and $\sigma_x^2 = 1$. For priors,

$$\begin{aligned}
&\alpha \sim \mathcal{U}(1, 3) \\
&b = (0.5, 25, 8), \; B = diag(0.025, 40, 2) \\
&c = 5, \; C = 40 \\
&d = 10, \; D = 9.
\end{aligned}$$

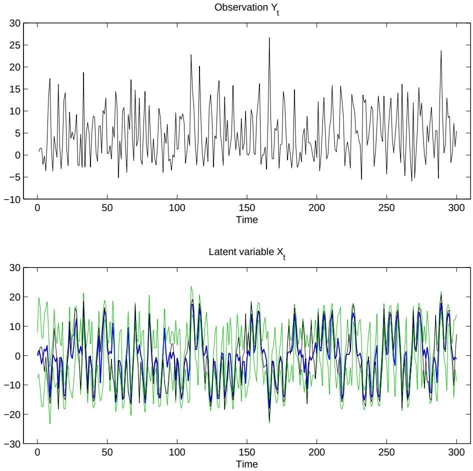
Data and States



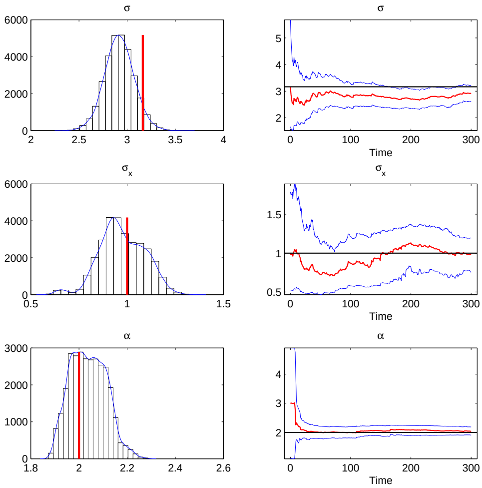
Sequential Learning



Posteriors


Observation Y_t
30
25
20
15
10
5
0
−5
−10
0
50
100
150
200
250
300
Time
Latent variable X_t
30
20
10
0
−10
−20
−30
0
50
100
150
200
250
300
Time

Data


σ
6000
4000
2000
0
2
2.5
3
3.5
4
σ
5
4
3
2
0
100
200
300
Time
σx
6000
4000
2000
0
0.5
1
1.5
σx
1.5
1
0.5
0
100
200
300
Time
α
3000
2000
1000
0
1.8
2
2.2
2.4
2.6
α
4
3
2
0
100
200
300
Time

Bivariate Radar Tracking

Example

- Original MC filter ($T = 20$). Akasi and Kumumato (1977).

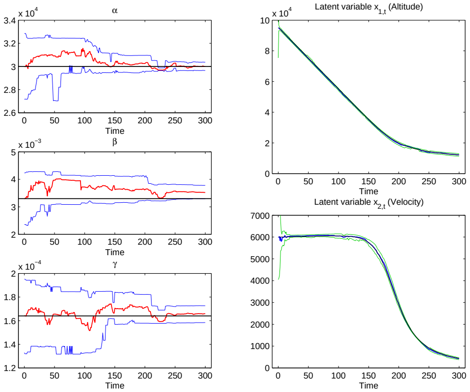
$$y_{t+1} = \sqrt{x_{1,t+1}^2 + \alpha^2} + \sigma(z_{t+1})\epsilon_{t+1}$$
$$x_{1,t+1} = x_{1,t} - x_{2,t}$$
$$x_{2,t+1} = x_{2,t} - \beta e^{-\gamma x_{1,t}} x_{2,t}^2 + g$$

  where $x_{1,t+1}, x_{2,t+1}$ and g are altitude (m), velocity (m/sec *downward*), and acceleration of gravity (m/sec^2).

- Parameters $\alpha = 30,000, \beta = 3.3 \times 10^{-3}, \gamma = -1.64 \times 10^{-4}$.

$$u_{t+1} \leq e^{-\frac{\left(y_{t+1} - \sqrt{x_{1,t+1}^2 + \alpha^2}\right)^2}{2\sigma(z_{t+1})^2}} \quad \text{and} \quad x_{1,t+1} = -\frac{1}{\gamma} \log\left(\frac{g - x_{2,t+1} + x_{2,t}}{\beta x_{2,t}^2}\right)$$

- Here $T = 300$ and parameter learning



α
x 10^4
3.4
3.2
3
2.8
2.6
0 50 100 150 200 250 300
Time
β
x 10^−3
5
4
3
2
0 50 100 150 200 250 300
Time
γ
x 10^−4
2
1.8
1.6
1.4
1.2
0 50 100 150 200 250 300
Time
Latent variable x1,t (Altitude)
x 10^4
10
8
6
4
2
0
0 50 100 150 200 250 300
Time
Latent variable x2,t (Velocity)
7000
6000
5000
4000
3000
2000
1000
0
0 50 100 150 200 250 300
Time

Discussion

- Particle Learning (CJLP, 2008)
  extends to nonlinear parameter models
- Marginalisation for efficiency
- Slice variable created conditional sufficient statistics
- Far more efficient than MCEM methods