Stat 535 C - Statistical Computing & Monte Carlo Methods

Lecture 22 - 30th March 2006

Arnaud Doucet

Email: arnaud@cs.ubc.ca

1.1– Outline

- Sequential Importance Sampling Resampling for Optimal Filtering.
- Limitations and Generalizations.
- Examples.

2.1– Nonlinear non-Gaussian State-space models

- Nonlinear non-Gaussian state-space model

$$X_1 \sim \mu, \;\; X_k | (X_{k-1} = x_{k-1}) \sim f(\cdot | x_{k-1}),$$

$$Y_k | (X_k = x_k) \sim g(\cdot | x_k).$$

- We are interested in the sequence of posterior distributions

$$\begin{aligned} p(x_{1:n} | y_{1:n}) &\propto p(x_{1:n})\, p(y_{1:n} | x_{1:n}) \\ &= \underbrace{\mu(x_1) \prod_{k=2}^{n} f(x_k | x_{k-1})}_{\text{prior}} \underbrace{\prod_{k=1}^{n} g(y_k | x_k)}_{\text{likelihood}}. \end{aligned}$$

2.2– Sequential Importance Sampling Resampling

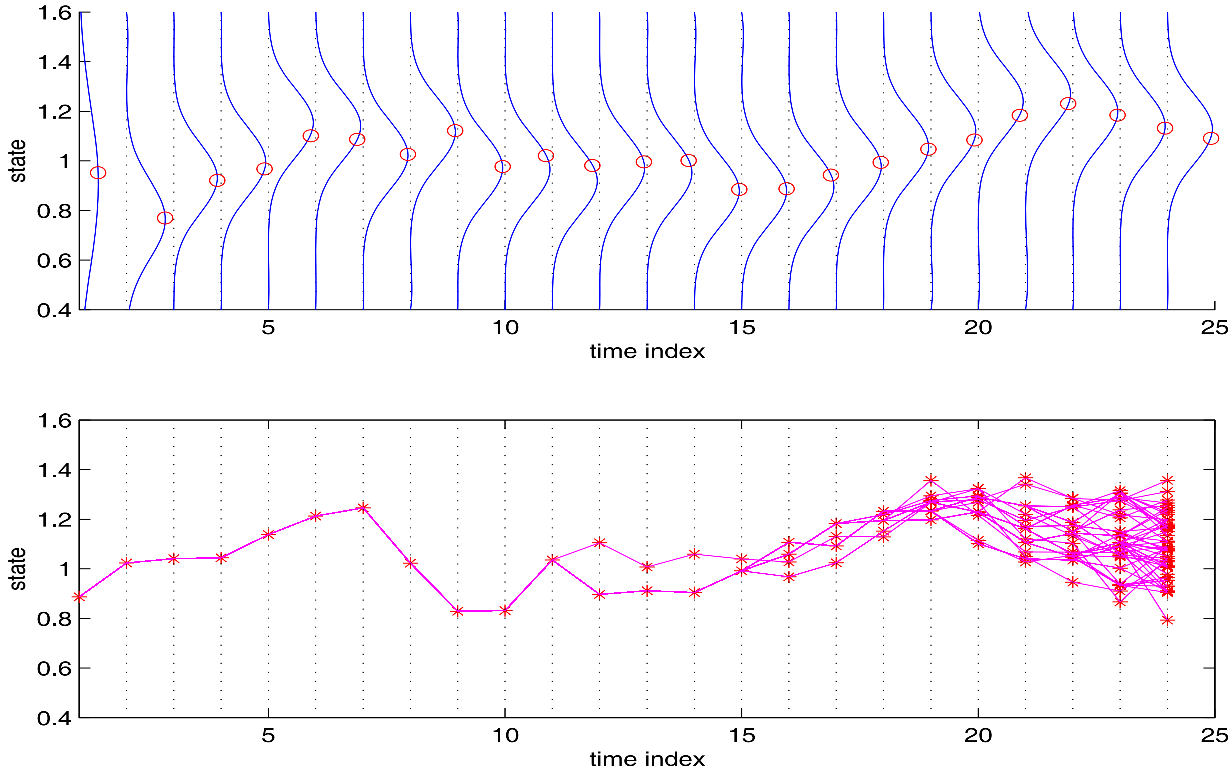
- At time n
  - Sample $X_n^{(i)} \sim q_n\left(x_n \middle| y_n, X_{n-1}^{(i)} \right)$ for $i = 1, ..., N$
  - Compute the weights

$$
\begin{aligned}
W_n^{(i)} &\propto W_{n-1}^{(i)} \frac{p\left(X_{1:n}^{(i)} \middle| y_{1:n}^{(i)} \right)}{p\left(X_{n-1}^{(i)} \middle| y_{1:n-1}^{(i)} \right) q_n\left(X_n^{(i)} \middle| y_n, X_{n-1}^{(i)} \right)} \\
&\propto W_{n-1}^{(i)} \frac{f\left(X_n^{(i)} \middle| X_{n-1}^{(i)} \right) g\left(y_n \middle| X_n^{(i)} \right)}{q_n\left(X_n^{(i)} \middle| y_n, X_{n-1}^{(i)} \right)}
\end{aligned}
$$

  - If the variation of the weights is high, resample the particles

$\left\{ X_{1:n}^{(i)}, W_n^{(i)} \right\}$ to obtain a new population $\left\{ X_{1:n}^{(i)}, 1/N \right\}$.

2.2– Sequential Importance Sampling Resampling



Particle Approximation of $p\left(x_{1:n}\middle|\, y_{1:n}\right)$. All the paths $X_{1:11}^{(i)}$ at time $n = 24$ have coalesce and $p\left(x_{1:1}\middle|\, y_{1:24}\right)$ is approximated using one single Delta mass.

• You can only expect to approximate the "most recent" marginals $p\left(x_{n-L+1:n}\middle|\, y_{1:n}\right)$ but NOT the joint distributions $p\left(x_{1:n}\middle|\, y_{1:n}\right)$ if the model has "mixing" properties, i.e. past errors are forgotten quickly.

• This seems rather limited but in most real-world applications we are only interested in the so-called filtering distribution $p\left(x_n\middle|\, y_{1:n}\right)$ and we can also use the following property to estimate smoothing distributions

$$p\left(x_k\middle|\, y_{1:n}\right) \simeq p\left(x_k\middle|\, y_{1:k+L}\right)$$

if the system has ergodic properties. Finally we have

$$p\left(y_{1:n}\right) = p\left(y_1\right) \prod_{k=2}^{n} p\left(y_k\middle|\, y_{1:k-1}\right)$$

where

$$p\left(y_k\middle|\, y_{1:k-1}\right) = \int g\left(y_k\middle|\, x_k\right) p\left(x_k\middle|\, y_{1:k-1}\right) dx_k.$$

2.3– A Nonlinear Example

$$X_{n+1} = \frac{X_n}{2} + 25\frac{X_n}{1+X_n^2} + 8\cos(1.2n) + \sigma_v V_{n+1} = \varphi(X_n) + V_{n+1},\ V_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 10).$$

$$Y_n = \frac{X_n^2}{20} + \sigma_w W_n = \Psi(X_n) + \sigma_w W_n,\ W_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1).$$

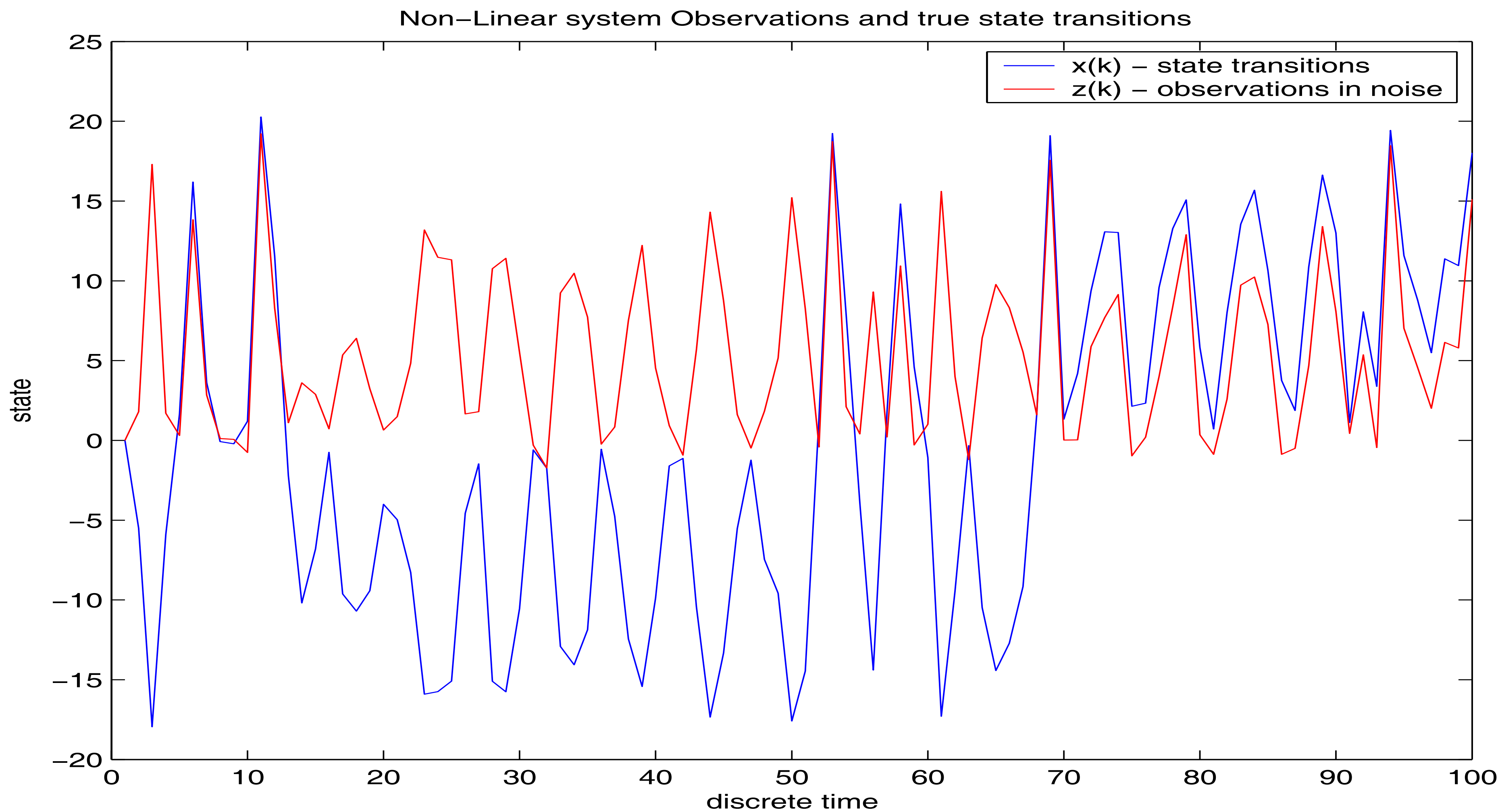
• Sampling from prior $\rightsquigarrow$ "Disastrous" as σ_v large/σ_w small and impossible to sample from $p(x_n | x_{n-1}, y_n)$.

• Local linearization

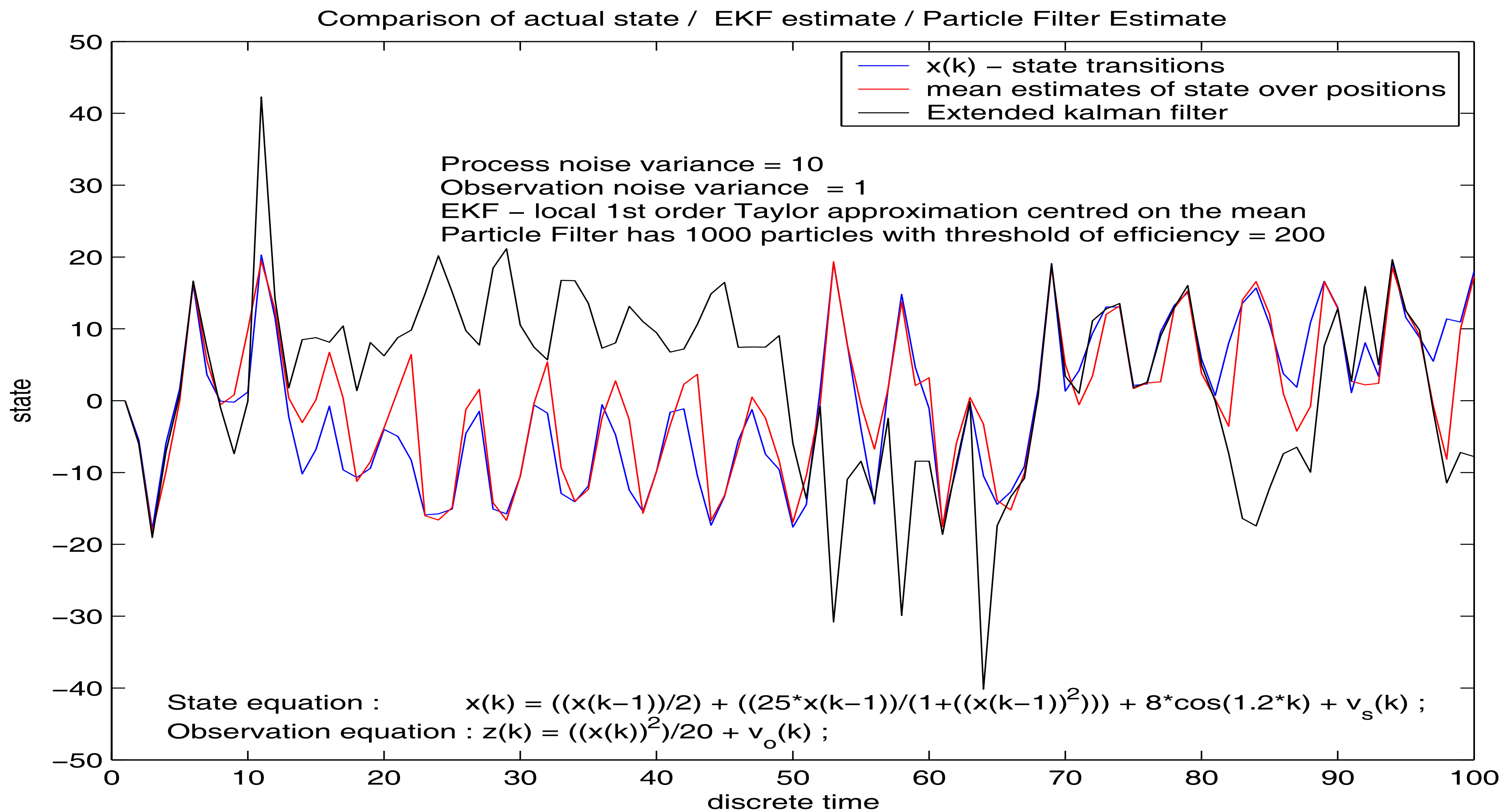
$$Y_n \approx \Psi(\varphi(X_{n-1})) + \nabla\Psi(\varphi(X_{n-1}))(X_n - \varphi(X_{n-1})) + \sigma_w W_n$$

$\Rightarrow$ Now in the form with linear observation...

2.3– A Nonlinear Example



2.3– A Nonlinear Example



3.1– Towards General SMC Methods

- The SMC approach can be extended to any sequence of target distributions

$$\pi_n(x_{1:n}) = \frac{\gamma_n(x_{1:n})}{Z_n}.$$

- In particular, we do not require the target distribution to satisfy

$$\pi_n(x_{1:n}) \propto \mu(x_1) \prod_{k=2}^{n} f(x_k | x_{k-1}) \prod_{k=1}^{n} g(y_k | x_k).$$

- The only requirement here is that

$$\pi_n(x_{1:n-1}) > 0 \Rightarrow \pi_{n-1}(x_{1:n-1}) > 0,$$

i.e. we can use $\pi_{n-1}(x_{1:n-1})$ to approximate $\pi_n(x_{1:n-1})$.

3.2– Conditionally Linear Gaussian State-Space Models

- As an example consider a switching state-space model

$$\begin{aligned} Z_n &= A(X_n) Z_{n-1} + B(X_n) V_n,\ Z_1 \sim \mathcal{N}(0, \Sigma_0),\ V_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I) \\ Y_n &= C(X_n) Z_n + D(X_n) W_n,\ W_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I) \end{aligned}$$

where X_n is an unobserved Markov process

$$X_1 \sim \mu, \quad X_n | X_{n-1} = x \sim f(\cdot | x).$$

- Ubiquitous class of models in CS, engineering, statistics and economics.

3.2– Conditionally Linear Gaussian State-Space Models

- *Finite Markov chain in colored & white noises* (ion channel): X_n finite Markov chain

$$Z_n = AZ_{n-1} + BV_n,\ Y_n = Z_n + X_n + W_n$$

- *Flat fading channels* (comms): X_n finite Markov chain

$$Z_n = AZ_{n-1} + BV_n, Y_n = Z_n X_n + W_n$$

- *Time Varying autoregressions in noise* (speech denoising)

$$X_n = X_{n-1} + E_n,$$

$$Z_n = A\left(X_n\right) Z_{n-1} + B\left(X_n\right) V_n,$$

$$Y_n = Y_{n-L+1:n}^{\mathsf{T}} Z_n + W_n.$$

- We could estimate using SMC

$$p\left(x_{1:n}, z_{1:n} \middle| y_{1:n}\right) \propto p\left(x_{1:n}\right) p\left(z_{1:n} \middle| x_{1:n}\right) p\left(y_{1:n} \middle| x_{1:n}, z_{1:n}\right)$$

$$= \mu\left(x_1\right) \prod_{k=2}^{n} f\left(x_k \middle| x_{k-1}\right) p\left(z_1\right) \prod_{k=2}^{n} \mathcal{N}\left(z_n; A\left(x_n\right) z_{n-1}, B\left(x_n\right) B^{\mathsf{T}}\left(\right.\right.$$

$$\times \prod_{k=1}^{n} \mathcal{N}\left(y_n; C\left(x_n\right) z_n, D\left(x_n\right) D^{\mathsf{T}}\left(x_n\right)\right)$$

- This fits in the framework discussed previously: $\{X_n, Z_n\}$ is a Markov process and the observations $\{Y_n\}$ are conditionally independent given $\{X_n, Z_n\}$.

- However, conditional upon $\{X_n\}$ the model is linear Gaussian. It follows that we have

$$p\left(x_{1:n}, z_{1:n} \middle| y_{1:n}\right) = p\left(x_{1:n} \middle| y_{1:n}\right) \underbrace{p\left(z_{1:n} \middle| y_{1:n}, x_{1:n}\right)}_{\text{Gaussian distribution}}$$

and it is only necessary to estimate through SMC the marginal distribution

$$p\left(x_{1:n} \middle| y_{1:n}\right) \propto p\left(y_{1:n} \middle| x_{1:n}\right) p\left(x_{1:n}\right)$$

where the likelihood term is given by the Kalman filter.

- We have $p\left(y_{1:n} \middle| x_{1:n}\right) \neq \prod_{k=1}^{n} p\left(y_k \middle| x_k\right)$ but this does not matter! Additionally we could have also a process $\{X_n\}$ which is non-Markovian. As long as we can compute the target up to a normalizing constant then we will be able to apply SMC.

3.4– Sequential Importance Sampling

- We can use sequential importance sampling

$$X_{1:n}^{(i)} \sim q_n(x_{1:n})$$

where

$$\begin{aligned} q_n(x_{1:n}) &= q_{n-1}(x_{1:n-1})\, q_n(x_n|\, y_{1:n}, x_{1:n-1}) \\ &= q_1(x_1) \prod_{k=2}^{n} q_k(x_k|\, x_{1:k-1}) \end{aligned}$$

- Whether the process is Markov or not does NOT matter whatsoever!

3.4– Sequential Importance Sampling

- We need to compute the weights associated to each particles $X_{1:n}^{(i)}$.

- We have

$$
\begin{aligned}
w_n(x_{1:n}) &= \frac{\gamma_n(x_{1:n})}{q_n(x_{1:n})} \propto \frac{\pi_n(x_{1:n})}{q_n(x_{1:n})} \\
&= \frac{\pi_{n-1}(x_{1:n-1})}{q_{n-1}(x_{1:n-1})} \frac{\pi_n(x_{1:n})}{\pi_{n-1}(x_{1:n-1})\, q_n(x_n | x_{1:n-1})} \\
&\propto w_{n-1}(x_{1:n-1}) \frac{\pi_n(x_{1:n})}{\pi_{n-1}(x_{1:n-1})\, q_n(x_n | x_{1:n-1})}.
\end{aligned}
$$

- In many cases, we can compute the incremental weight in a computational time independent of n. If not the computational complexity increases with n.

3.5– Selection of the Importance Distribution

- We propose to select the importance distribution minimizing the variance of the incremental importance weight conditional upon $x_{1:n-1}$.

- We have

$$\begin{aligned} w_n(x_{1:n}) &\propto w_{n-1}(x_{1:n-1}) \frac{\pi_n(x_{1:n})}{\pi_{n-1}(x_{1:n-1}) q_n(x_n|x_{1:n-1})} \\ &\propto w_{n-1}(x_{1:n-1}) \frac{\pi_n(x_{1:n-1})}{\pi_{n-1}(x_{1:n-1})} \frac{\pi_n(x_n|x_{1:n-1})}{q_n(x_n|x_{1:n-1})} \end{aligned}$$

so the (locally) optimal choice is

$$q_n(x_n|x_{1:n-1}) = \pi_n(x_n|x_{1:n-1})$$

3.6– Sequential Importance Sampling Resampling

- At time n
  - Sample $X_n^{(i)} \sim q_n\left(x_n \middle| X_{1:n-1}^{(i)}\right)$ for $i = 1, ..., N$
  - Compute the weights

$$W_n^{(i)} \propto W_{n-1}^{(i)} \frac{\pi_n\left(X_{1:n}^{(i)}\right)}{\pi_{n-1}\left(X_{1:n-1}^{(i)}\right) q_n\left(X_n^{(i)} \middle| X_{1:n-1}^{(i)}\right)}$$

  - If the variation of the weights is high, resample the particles

$\left\{X_{1:n}^{(i)}, W_n^{(i)}\right\}$ to obtain a new population $\left\{X_{1:n}^{(i)}, 1/N\right\}$.

3.7– Application of SMC to Conditionally Linear Gaussian Models

• In this case, the target is $\pi_n(x_{1:n}) = p(x_{1:n}|\, y_{1:n})$ and the optimal importance distribution is given by

$$\begin{aligned}\pi_n(x_n|\, x_{1:n-1}) &= p(x_n|\, y_{1:n}, x_{1:n-1}) \\ &\propto p(y_n|\, y_{1:n-1}, x_{1:n})\, f(x_n|\, x_{n-1})\end{aligned}$$

• For an importance distribution, importance weight proportional to

$$\begin{aligned}w_n(x_{1:n}, y_{1:n}) &\propto \frac{p(x_{1:n}|\, y_{1:n})}{p(x_{1:n}|\, y_{1:n})\, q(x_n|\, y_{1:n}, x_{1:n-1})} \\ &\propto \frac{p(y_n|\, y_{1:n-1}, x_{1:n})\, f(x_n|\, x_{n-1})}{q(x_n|\, y_{1:n}, x_{1:n-1})}.\end{aligned}$$

3.8– SISR for conditionally linear Gaussian state space

At time $n-1$, $\left\{W_{n-1}^{(i)}, X_{1:n-1}^{(i)}\right\}$

- *Sampling Step.* For $i = 1, ..., N$, sample $X_n^{(i)} \sim q_n\left(\cdot|\, X_{1:n-1}^{(i)}, y_{1:n}\right)$

$$W_n^{(i)} \propto W_{n-1}^{(i)} \frac{p\left(y_n|\, y_{1:n-1}, X_{1:n}^{(i)}\right) f\left(X_n^{(i)}\middle|\, X_{n-1}^{(i)}\right)}{q\left(X_n^{(i)}\middle|\, y_{1:n}, X_{1:n-1}^{(i)}\right)}$$

- *Resampling Step.* If variance of weights $\left\{W_n^{(i)}\right\}$ high, resample $\left\{W_n^{(i)}, X_{1:n}^{(i)}\right\}$ to obtain $\left\{N^{-1}, X_{1:n}^{(i)}\right\}$.

- At first glance, memory requirements increase with n even if one focusses on $p\left(X_n|\, y_{1:n}\right)$.

• At first glance, this algorithm is not practical because we need to keep in memory $X_{1:n}^{(i)}$ to compute $w_n(x_{1:n}, y_{1:n})$ and a "good" importance distribution should depend on $p(y_n | y_{1:n-1}, x_{1:n})$.

• However, we have

$$p(y_n | y_{1:n-1}, x_{1:n}) = \int g(y_n | z_n, x_n) f(z_n | z_{n-1}) p(z_{n-1} | y_{1:n-1}, x_{1:n-1}) dz_{n-1:n}$$

where $p(z_{n-1} | y_{1:n-1}, x_{1:n})$ is a Gaussian distribution whose mean $m_{n-1}(x_{1:n-1})$ and covariance $\Sigma_{n-1}(x_{1:n-1})$ can be computed by the Kalman filter as the model is conditionally linear Gaussian.

• In practice, we do NOT need to store $X_{1:n}^{(i)}$ but only $\left(X_n^{(i)}, m_n\left(X_{1:n}^{(i)}\right), \Sigma_n\left(X_{1:n}^{(i)}\right)\right)$ denoted $\left(X_n^{(i)}, m_n^{(i)}, \Sigma_n^{(i)}\right)$, i.e. $\left(X_n^{(i)}, m_n^{(i)}, \Sigma_n^{(i)}\right)$ are sufficient statistics summarizing all the information about $\left(X_{1:n}^{(i)}, y_{1:n}\right)$.

3.10– "Real" Implementation of SISR

At time $n-1$, $\left\{W_{n-1}^{(i)}, X_{n-1}^{(i)}, m_{n-1}^{(i)}, \Sigma_{n-1}^{(i)}\right\}$

- *Sampling Step.* For $i = 1, ..., N$, sample $\beta_n^{(i)} \sim q\left(\cdot \middle| y_n, X_{n-1}^{(i)}, m_{n-1}^{(i)}, \Sigma_{n-1}^{(i)}\right)$

$$W_n^{(i)} \propto W_{n-1}^{(i)} \frac{p\left(y_n \middle| X_{n-1}^{(i)}, m_{n-1}^{(i)}, \Sigma_{n-1}^{(i)}\right) f\left(X_n^{(i)} \middle| X_{n-1}^{(i)}\right)}{q\left(X_n^{(i)} \middle| y_n, X_{n-1}^{(i)}, m_{n-1}^{(i)}, \Sigma_{n-1}^{(i)}\right)}$$

- *Kalman Step.* For $i = 1, ..., N$, use the Kalman filter to compute $\left(m_n^{(i)}, \Sigma_n^{(i)}\right)$.

- *Resampling Step.* If variance of weights $\left\{W_n^{(i)}\right\}$ high, resample $\left\{W_n^{(i)}, m_n^{(i)}, \Sigma_n^{(i)}, X_n^{(i)}\right\}$

to obtain $\left\{N^{-1}, m_n^{(i)}, \Sigma_n^{(i)}, X_n^{(i)}\right\}$.

3.10– "Real" Implementation of SISR

- Computational complexity: N Kalman filters.

- Inference on X_n based on $\widehat{p}(x_n | y_{1:n}) = \sum_{i=1}^{N} W_n^{(i)} \delta_{X_n^{(i)}}(x_n)$.

- Inference on α_n based on

$$\widehat{p}(z_n | y_{1:n}) = \int p(z_n | y_{1:n}, x_{1:n}) \widehat{p}(dx_{1:n} | y_{1:n}) = \sum_{i=1}^{N} W_n^{(i)} \mathcal{N}\left(z_n; m_n^{(i)}, \Sigma_n^{(i)}\right).$$

- One has

$$\widehat{E}(Z_n | y_{1:n}) = \sum_{i=1}^{N} W_n^{(i)} m_n^{(i)},$$

$$\widehat{\text{cov}}(Z_n | y_{1:n}) = \sum_{i=1}^{N} W_n^{(i)} \left(\Sigma_n^{(i)} + m_n^{(i)} m_n^{(i)\mathtt{T}}\right) - \widehat{E}(Z_n | y_{1:n}) \widehat{E}^{\mathrm{T}}(Z_n | y_{1:n})$$

3.11– Applications to Deconvolution of Bernoulli-Gauss Processes

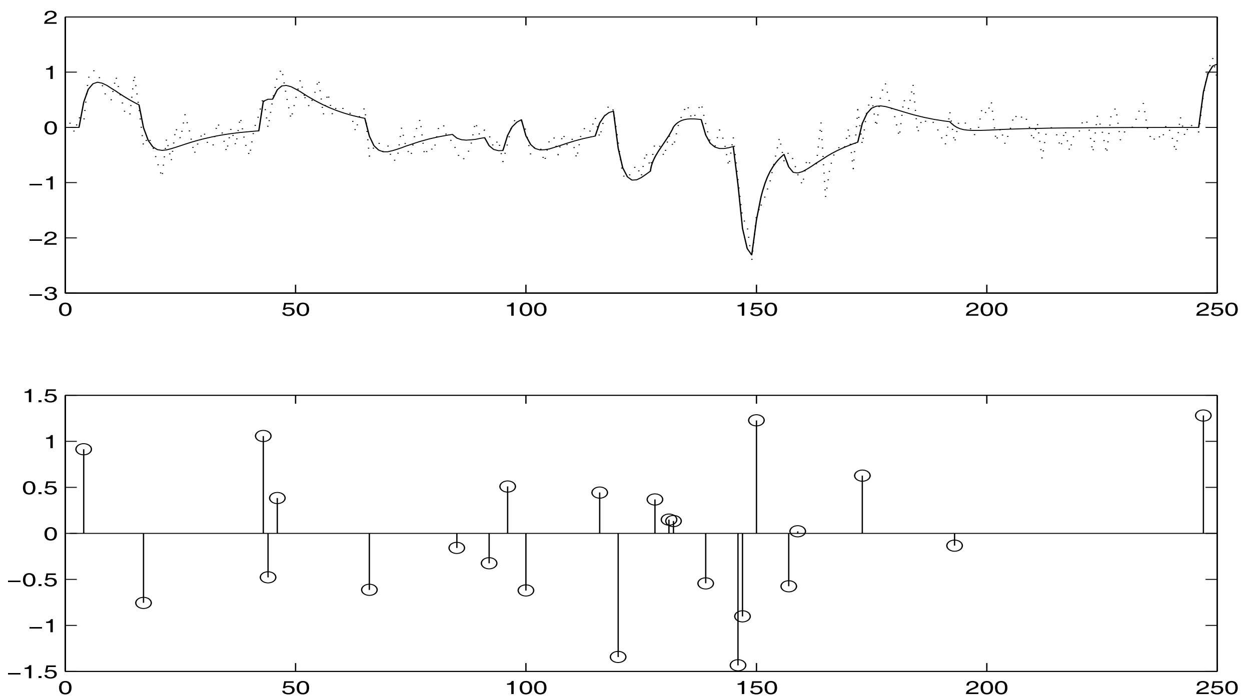
- Bernoulli-Gauss input sequence for an AR(2) filter

$$V_n' \overset{\text{i.i.d.}}{\sim} \lambda \mathcal{N}\left(0, \sigma_v^2\right) + (1-\lambda)\,\delta_0, 0 < \lambda < 1$$

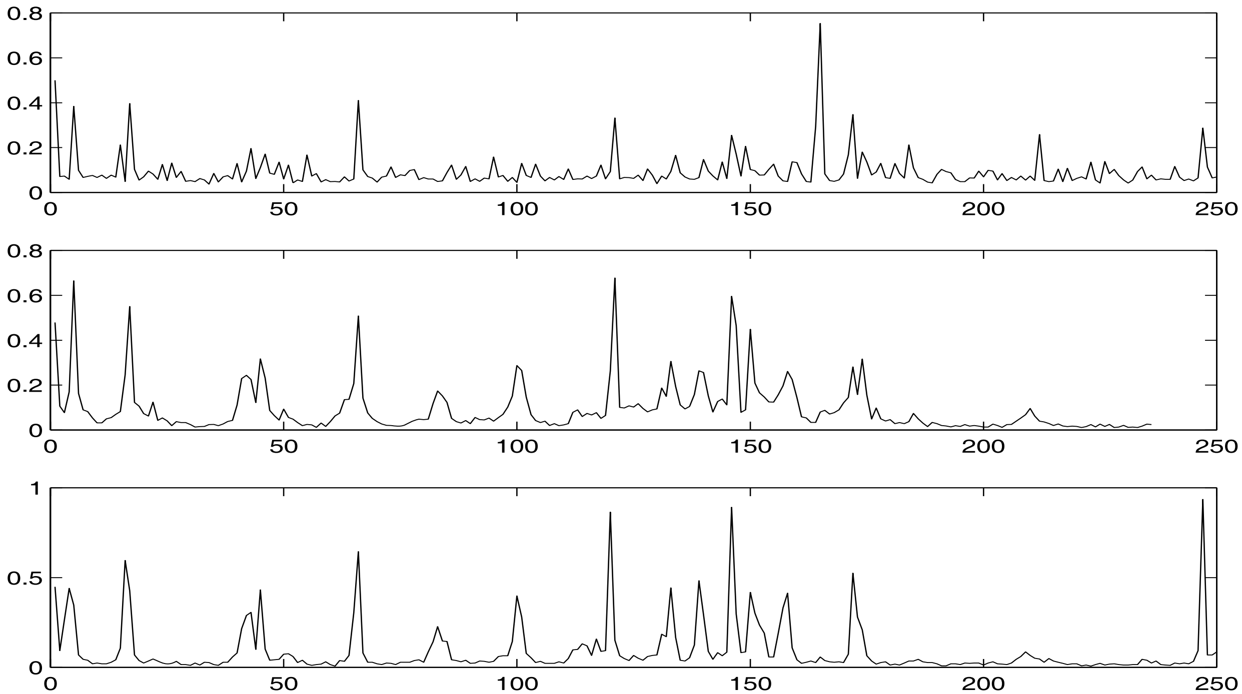
- Jump Markov system: $A = \begin{pmatrix} a_1 & a_2 \\ 1 & 0 \end{pmatrix}$, $C = \begin{pmatrix} 1 & 0 \end{pmatrix}$, $D = \sigma_w$, $B(1) = \begin{pmatrix} \sigma_v & 0 \end{pmatrix}^{\mathtt{T}}$, $B(2) = \begin{pmatrix} 0 & 0 \end{pmatrix}^{\mathtt{T}}$.

- Parameters to $a_1 = 1.51$, $a_2 = -0.55$, $\sigma_v = 0.50$ and $\sigma_w = 0.25$ and $N = 500$ particles, fixed-lag smoothing $L = 15$.

Top: $\{X_n\}$ (solid) and $\{Y_n\}$ (dashed). Bottom: Input sequence $\{V'_n\}$



Top: Estimates of $\Pr(V'_n \neq 0 | y_{1:n})$, Middle: Estimates of $\Pr(V'_n \neq 0 | y_{1:n+15})$ and Bottom: Estimates of $\Pr(V'_n \neq 0 | y_{1:T})$ with Gibbs sampling.



3.12– Partially observed linear Gaussian state space models

$$Z_{n+1} = AZ_n + BV_{n+1},\ V_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, I_{n_v}\right),\ X_n = CZ_n + DW_n,\ W_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, I_{n_w}\right)$$

$$Y_n | \left(Z_n, X_n\right) \sim g\left(\cdot | X_n\right)$$

• One can use SMC on $p\left(\alpha_{1:n} | y_{1:n}\right)$ where $\alpha = \left(z_n, x_n\right)$ but inefficient as

$$p\left(z_{1:n}, x_{1:n} | y_{1:n}\right) = p\left(x_{1:n} | y_{1:n}\right) \underbrace{p\left(z_{1:n} | x_{1:n}\right)}_{\text{Gaussian}}$$

• The distribution $p\left(x_{1:n} | y_{1:n}\right)$ can be computed analytically up to a normalizing constant

$$p\left(x_{1:n} | y_{1:n}\right) \propto \underbrace{p\left(y_{1:n} | x_{1:n}\right)}_{\text{Likelihood}} \cdot \underbrace{p\left(x_{1:n}\right)}_{\text{Prior computed by Kalman}}$$

$\Rightarrow$ SMC on $\pi_n\left(x_{1:n}\right) = p\left(x_{1:n} | y_{1:n}\right)$.

- *Quantized observations* (speech processing, radar, finance)
- *Dynamic Tobit Model* (econometrics)

$$\begin{aligned} Z_n &= \phi Z_{n-1} + \sigma_v V_n,\ X_n = Z_n + \sigma_w W_n, \\ Y_n &= 0 \vee X_n. \end{aligned}$$

- *Dynamic Probit Model* (econometrics)

$$\begin{aligned} &Z_n = \phi Z_{n-1} + \sigma_v V_n, \\ &\Pr\left(Y_n = 1 | Z_n \right) = \Phi\left(Z_n \right) \end{aligned} \Rightarrow \begin{aligned} &Z_n = \phi Z_{n-1} + \sigma_v V_n, \\ &X_n = Z_n + W_n,\ W_n \sim \mathcal{N}\left(0,1\right) \\ &Y_n = \mathbb{I}_{R^+}\left(X_n \right). \end{aligned}$$

3.14– Application of SMC

• Select an importance distribution $q\left(x_n \mid x_{1:n-1}, y_{1:n}\right)$... Optimal importance distribution

$$q\left(x_n \mid x_{1:n-1}, y_{1:n}\right) = p\left(x_n \mid x_{1:n-1}, y_{1:n}\right) \propto g\left(y_n \mid x_n\right) p\left(x_n \mid x_{1:n-1}\right)$$

• Importance weight

$$\frac{\pi_n\left(x_{1:n}\right)}{\pi_{n-1}\left(x_{1:n-1}\right) q\left(x_n \mid x_{1:n-1}, y_{1:n}\right)} \propto \frac{g\left(y_n \mid x_n\right) p\left(x_n \mid x_{1:n-1}\right)}{q\left(x_n \mid x_{1:n-1}, y_{1:n}\right)}.$$

3.15– SISR for partially observed Gaussian state space

At time $n-1$, $\left\{W_{n-1}^{(i)}, X_{1:n-1}^{(i)}\right\}$

- *Sampling Step.* For $i = 1, ..., N$, sample $X_n^{(i)} \sim q_n\left(\cdot \mid X_{1:n-1}^{(i)}, y_{1:n}\right)$

$$W_n^{(i)} \propto W_{n-1}^{(i)} \frac{g\left(y_n \mid X_n^{(i)}\right) p\left(X_n^{(i)} \middle| X_{1:n-1}^{(i)}\right)}{q\left(X_n^{(i)} \middle| X_{1:n-1}^{(i)}, y_{1:n}\right)}$$

- *Selection Step.* If variance of weights $\left\{W_n^{(i)}\right\}$ high, resample $\left\{W_n^{(i)}, X_{1:n}^{(i)}\right\}$ to obtain $\left\{N^{-1}, X_{1:n}^{(i)}\right\}$.

- At first glance, memory requirements increase with n even if one focusses on $p\left(x_n \mid y_{1:n}\right)$.

3.15– SISR for partially observed Gaussian state space

- Like for conditionally Gaussian models, the key is to notice that

$$p(x_n|x_{1:n-1}) = \int p(x_n|z_n) f(z_n|z_{n-1}) p(z_{n-1}|x_{1:n-1}) dz_{n-1:n}$$

where $p(z_{n-1}|x_{1:n-1})$ is a Gaussian distribution whose mean m_{n-1} and covariance Σ_{n-1} is computed through the Kalman filter associated to the "virtual" observations.

- In practice, we do NOT need to store $X_{1:n}^{(i)}$ but only $\left(X_n^{(i)}, m_n\left(X_{1:n}^{(i)}\right), \Sigma_n\left(X_{1:n}^{(i)}\right)\right)$ denoted $\left(X_n^{(i)}, m_n^{(i)}, \Sigma_n^{(i)}\right)$, i.e. $\left(X_n^{(i)}, m_n^{(i)}, \Sigma_n^{(i)}\right)$ are sufficient statistics summarizing all the information about $\left(X_{1:n}^{(i)}\right)$.

At time $n-1$, $\left\{W_{n-1}^{(i)}, m_{n-1}^{(i)}, X_{n-1}^{(i)}\right\}$ and $\Sigma_{n-1}^{(i)} = \Sigma_{n-1}$.

- *Sampling Step.* For $i = 1, ..., N$, sample $X_n^{(i)} \sim q\left(\cdot|\, y_n, m_{n-1}^{(i)}, \Sigma_{n-1}\right)$

$$W_n^{(i)} \propto \frac{g\left(y_n|\, X_n^{(i)}\right) p\left(X_n^{(i)}\middle|\, m_{n-1}^{(i)}, \Sigma_{n-1}\right)}{q\left(X_n^{(i)}\middle|\, y_n, m_{n-1}^{(i)}, \Sigma_{n-1}\right)}$$

- *Kalman Step.* For $i = 1, ..., N$, use the Kalman filter to compute $\left\{m_n^{(i)}\right\}$ and Σ_n.

- *Selection Step.* If variance of weights $\left\{W_n^{(i)}\right\}$ high, resample $\left\{W_n^{(i)}, m_{n-1}^{(i)}, X_{n-1}^{(i)}\right\}$ to obtain $\left\{N^{-1}, m_{n-1}^{(i)}, X_{n-1}^{(i)}\right\}$.

3.17– Using the RB SMC Filter

- Computational complexity $\neq N$ Kalman filters as $\Sigma_{n-1}^{(i)}$ independent of i.
- Inference on X_n based on $\widehat{p}\left(x_n \mid y_{1:n}\right)=\sum_{i=1}^N W_n^{(i)} \delta_{X_n^{(i)}}\left(x_n\right)$.
- Inference on Z_n based on

$$\widehat{p}\left(z_n \mid y_{1:n}\right)=\int p\left(z_n \mid x_{1:n}\right) p\left(d x_{1:n} \mid y_{1:n}\right)=\sum_{i=1}^N W_n^{(i)} \mathcal{N}\left(z_n ; m_n^{(i)}, \Sigma_n\right).$$

- One has

$$\widehat{E}\left(Z_n \mid y_{1:n}\right)=\sum_{i=1}^N W_n^{(i)} m_n^{(i)},$$

$$\widehat{\operatorname{cov}}\left(Z_n \mid y_{1:n}\right)=\sum_{i=1}^N W_n^{(i)}\left(\Sigma_n+m_n^{(i)} m_n^{(i) \mathrm{T}}\right)-\widehat{E}\left(Z_n \mid y_{1:n}\right) \widehat{E}^{\mathrm{T}}\left(Z_n \mid y_{1:n}\right)$$

3.18– Dynamic Tobit Model

State-space model (Manrique and Shephard, 1998)

$$Z_n = \phi Z_{n-1} + \sigma_v V_n, \; X_n = Z_n + \sigma_w W_n,$$

$$Y_n = 0 \vee X_n.$$

Importance distribution

- If $y_n > 0$, $y_n = X_n$ and $w_n(x_{1:n}) \propto w_{n-1}(x_{1:n-1}) \mathcal{N}\left(x_n; m_{n|n-1}, \Sigma_{x,\,n|n-1}\right)$.

- If $y_n = 0$, $q(x_n | x_{1:n-1}, y_{1:n}) \propto p(x_n | x_{1:n-1}) \mathbb{I}_{R^+}(x_n)$ and $w_n(x_{1:n}) \propto w_{n-1}(x_{1:n-1}) \Phi\left(-x_{n|n-1}/\sqrt{\Sigma_{x,\,n|n-1}}\right)$.

3.19– Monte Carlo Simulation

- Comparison with standard filter $p(z_n, x_n | y_n, z_{n-1}, x_{n-1})$ in terms of MSE for $M = 25$ realizations, $T = 100$.

Algorithm/N	50	100	250	500	1000	2500	5000
Standard Filter	543.07	540.41	538.77	538.02	536.90	533.33	533.76
MC Filter	534.20	533.46	533.29	533.12	533.21	533.18	533.20

- For a fixed N, this method is 2 times more computationally expensive but similar performance with 25-50 times less particles.

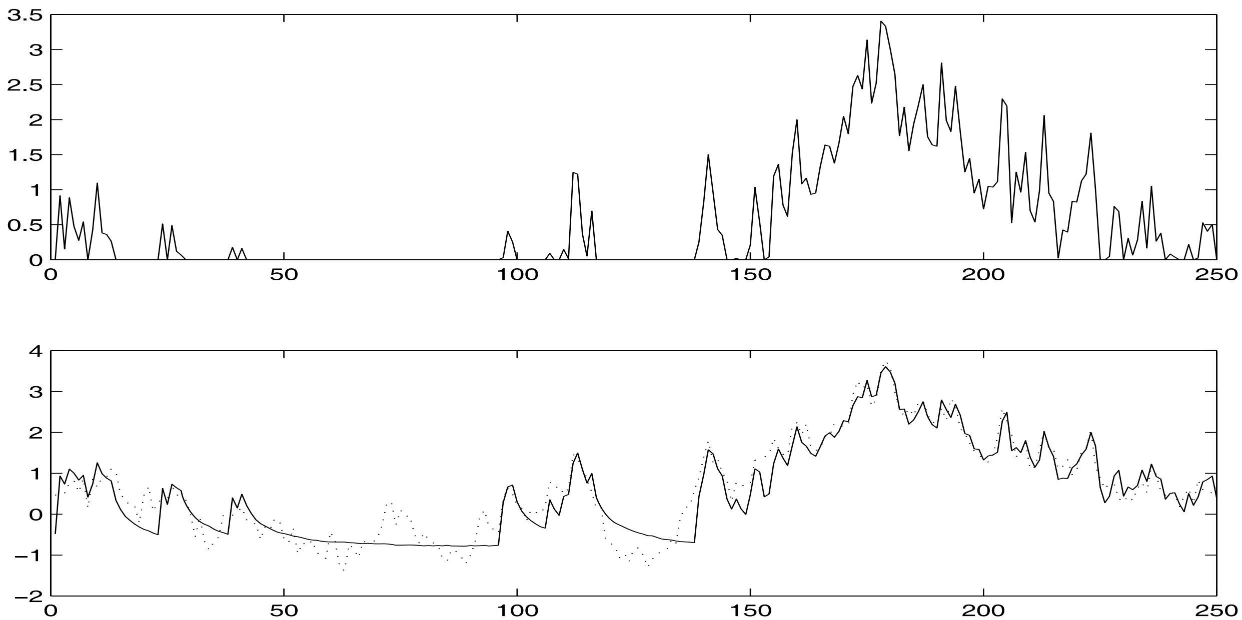


Figure 1: Simulated observations (solid line) (top), state (dashed line) (bottom) and MMSE state estimate via particle filter (solid line)

3.21– Dynamic Tobit Model

State-space model (Manrique and Shephard, 1998)

$$\gamma_{n+1} = 2\gamma_n - \gamma_{n-1} + \sigma_v V_{n+1} \text{ (state space } Z_n = (\gamma_n, \gamma_{n-1})\text{)}, \; X_n = \gamma_n + W_n,$$

$$Y_n = \mathbb{I}_{R^+}(X_n) \,.$$

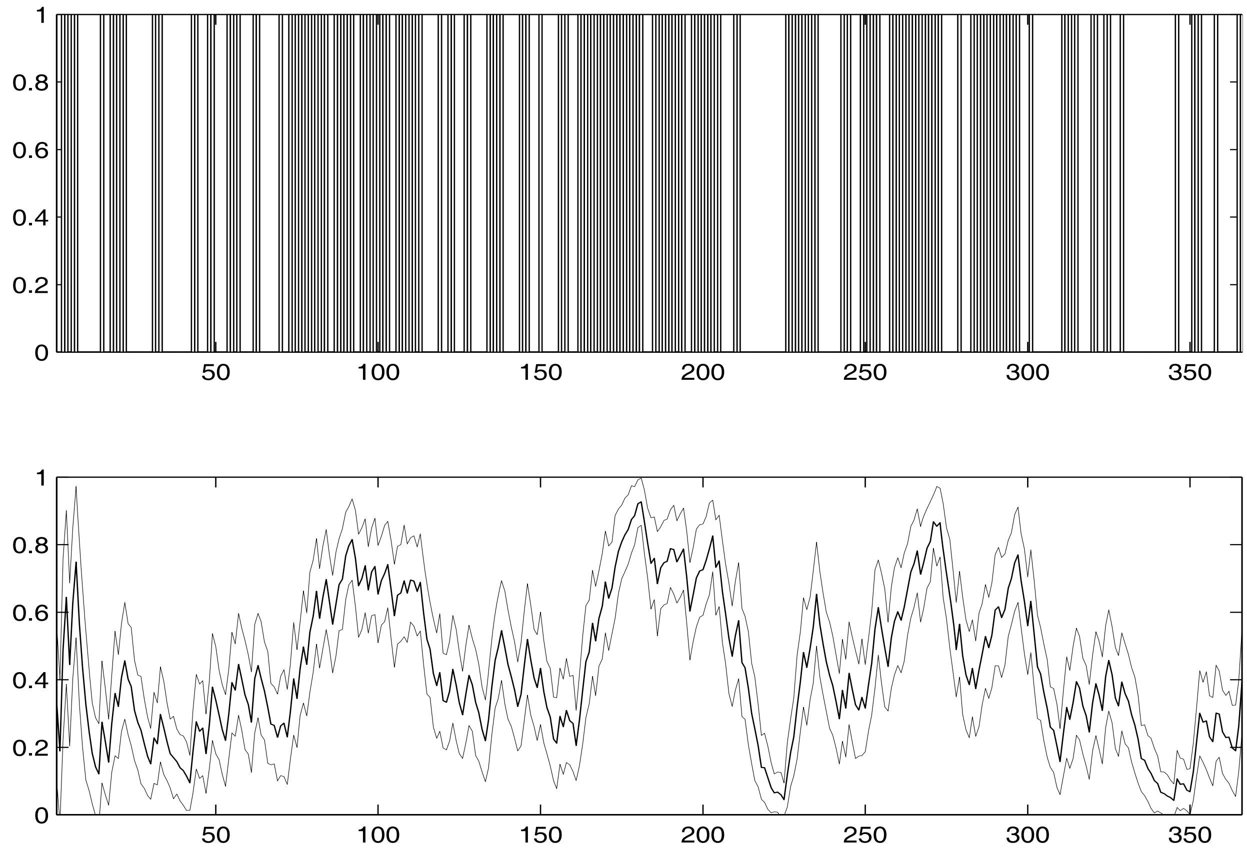
Importance distribution

- If $y_n = 1$, $q(x_n | x_{1:n-1}, y_{1:n}) \propto p(x_n | x_{1:n-1}) \mathbb{I}_{R^+}(x_n)$,
$w_n(x_{1:n}) \propto w_{n-1}(x_{1:n-1}) \left(1 - \Phi\left(-x_{n|n-1} / \sqrt{\Sigma_{x,\, n|n-1}}\right)\right)$.

- If $y_n = 0$, $q(x_n | x_{1:n-1}, y_{1:n}) \propto p(x_n | x_{1:n-1}) \mathbb{I}_{R^-}(x_n)$,
$w_n(x_{1:n}) \propto w_{n-1}(x_{1:n-1}) \, \Phi\left(-x_{n|n-1} / \sqrt{\Sigma_{x,\, n|n-1}}\right)$.

Introduction of latent variable. If one uses no latent variable and uses $p(z_n | z_{n-1}, y_n)$ then $p(y_n | z_{n-1})$ has no expression.

3.22– Tokyo Rainfall Data



4.1– Summary

- Sequential Monte Carlo is a simple and efficient methodology for sequential Bayesian inference.

- However, it remains highly limited: few degrees of freedom, how to deal with static problems?

- Next week we will show how to extend SMC to general problems.