Particle Smoothing

Introduction to Smoothing

- Particle filters (PFs) provide a very good approximation of $p_\theta(x_n|y_{0:n})$
  - in this case path degeneracy does not matter
- Path degeneracy means simple PFs **do not** provide good approximations for:
  - smoothers $p(x_n|y_{0:T})$ $(T > n)$
  - joint filtering $p(x_{0:n}|y_{0:n})$ and
  - smoothed additive functionals

$$\mathcal{S}_n^\theta = \int \left[\sum_{k=0}^{n} s_k\left(x_k, x_{k-1}\right) \right] p_\theta\left(x_{0:n} \middle| \, y_{0:n}\right) dx_{0:n},$$

Introduction to Smoothing

- Why is this important?
  - infering x_0 if unknown
  - backtracking events using $p(x_n|y_{0:T})$
    - when did a storm start to pick up?
  - **Parameter inference** and MLE estimation: EM and gradient algorithms for θ are of the form

$$\theta_k = \Lambda\left(\mathcal{S}_T^{\theta_{k-1}}\right)$$

with Λ an appropriate function

Additive functionals $\mathcal{S}_n^\theta$

- When estimating

$$\mathcal{S}_n^\theta = \int \left[\sum_{k=0}^{n} s_k \left(x_k , x_{k-1} \right) \right] p_\theta \left(\left. x_{0:n} \right| y_{0:n} \right) dx_{0:n}, \tag{1}$$

the theory tells that the Monte Carlo variance of simple PFs

$$\widehat{\mathcal{S}_n^\theta} = \int \left[\sum_{k=0}^{n} s_k \left(x_k , x_{k-1} \right) \right] \widehat{p}_\theta \left(\left. dx_{0:n} \right| y_{0:n} \right) , \tag{2}$$

satisfies

$$\mathbb{V}ar \left(\widehat{\mathcal{S}_n^\theta} \right) \geq D_\theta \frac{n^2}{N}. \tag{3}$$

even with exponential filter stability for HMM.

- This motivates the use of dedicated smoothing algorithms

Smoothing algorithms

- We are interested in designing better particle approximations for $p_\theta\left(x_{0:T} \mid y_{0:T}\right)$ and $\left\{p_\theta\left(x_n \mid y_{0:T}\right)\right\}_{n=0}^{T}$
- Some popular approaches
  - fixed lag smoothing
  - forward filtering backward sampling
  - forward filtering backward smoothing

Fixed lag smoothing

- For state-space models with "good" forgetting properties if L large enough then

$$p_\theta\left(x_{0:n}\middle|\, y_{0:T}\right) \approx p_\theta\left(x_{0:n}\middle|\, y_{0:(n+L)\wedge T}\right)$$

- observations collected at times $k > n + L$ do not bring any significant additional information about $X_{0:n}$.
- Fixed lag approximation (Kitagawa & Sato 2001):
  - do not resample the components $X^i_{0:n}$ of the particles $X^i_{0:k}$ obtained by particle filtering at times $k > n + L$.
- Could work in practice, but method is asymptotically biased and it might be hard to tune L.

Forward-Backward Smoothing using sampling

- Backward interpretation
- The joint smoothing distribution $p_\theta\left(x_{0:T} \mid y_{0:T}\right)$ can be expressed as a function of the filtering distributions $\left\{p_\theta\left(x_n \mid y_{0:n}\right)\right\}_{n=0}^{T}$ as follows

$$p_\theta\left(x_{0:T} \mid y_{0:T}\right) = p_\theta\left(x_T \mid y_{0:T}\right) \prod_{n=0}^{T-1} p_\theta\left(x_n \mid y_{0:n}, x_{n+1}\right) \tag{4}$$

where

$$p_\theta\left(x_n \mid y_{0:n}, x_{n+1}\right) = \frac{f_\theta\left(x_{n+1} \mid x_n\right) p_\theta\left(x_n \mid y_{0:n}\right)}{p_\theta\left(x_{n+1} \mid y_{0:n}\right)}. \tag{5}$$

Particle Implementation

- Forward Filtering Backward Sampling (FFBSa) :

  - Run a particle filter from time $n = 0$ to T, storing the approximate filtering distributions $\{\widehat{p}_\theta(\, dx_n |\, y_{0:n})\}_{n=0}^{T}$
  - Sample $\mathtt{X}_T \sim \widehat{p}_\theta(\, dx_T |\, y_{0:T})$ and
  - for $n = T-1, T-2, \ldots, 0$ sample

    $$\mathtt{X}_n \sim \widehat{p}_\theta(\, dx_n |\, y_{0:n}, \mathtt{X}_{n+1})$$

    where this distribution is obtained by substituting $\widehat{p}_\theta(\, dx_n |\, y_{0:n})$ for $p_\theta(\, dx_n |\, y_{0:n})$ in (5):

    $$\widehat{p}_\theta(\, dx_n |\, y_{0:n}, X_{n+1}) = \frac{\sum_{i=1}^{N} W_n^i f_\theta(\mathtt{X}_{n+1} | X_n^i) \delta_{X_n^i}(dx_n)}{\sum_{i=1}^{N} W_n^i f_\theta(\mathtt{X}_{n+1} | X_n^i)}. \quad (6)$$

Forward-Backward Smoothing

- A backward in time recursion for $\{p_\theta(x_n | y_{0:T})\}_{n=0}^{T}$ follows by integrating out $x_{0:n-1}$ and $x_{n+1:T}$ in (4) while applying (5):

$$
\begin{aligned}
p_\theta(x_n | y_{0:T}) &= \int p_\theta(x_n, x_{n+1} | y_{0:T})\, dx_{n+1} \\
&= \int p_\theta(x_n | y_{0:n}, x_{n+1})\, p_\theta(x_{n+1} | y_{0:T})\, dx_{n+1} \\
&= \int \frac{f_\theta(x_{n+1} | x_n)\, p_\theta(x_n | y_{0:n})}{p_\theta(x_{n+1} | y_{0:n})} p_\theta(x_{n+1} | y_{0:T})\, dx_{n+1}.
\end{aligned}
$$

Forward-Backward Smoothing

- So the backward in time recursion for $\{p_\theta(x_n|y_{0:T})\}_{n=0}^{T}$ is:

$$p_\theta(x_n|y_{0:T}) = p_\theta(x_n|y_{0:n}) \int \frac{f_\theta(x_{n+1}|x_n)\, p_\theta(x_{n+1}|y_{0:T})}{p_\theta(x_{n+1}|y_{0:n})} dx_{n+1}. \tag{7}$$

- So $\{p_\theta(x_n|y_{0:n})\}_{n=0}^{T}$ can be used in a backward pass to obtain $\{p_\theta(x_n|y_{0:T})\}_{n=0}^{T}$ and $\{p_\theta(x_n|y_{0:n}, x_{n+1})\}_{n=0}^{T-1}$.

Particle Implementation

- Forward Filtering Backward Smoothing (FFBSm) :
- Assume we have an approximation

$$\widehat{p}_{\theta}\left(dx_{n+1} \middle| y_{0:T} \right) = \sum_{i=1}^{N} W_{n+1|T}^{i} \delta_{X_{n+1}^{i}}(dx_{n+1})$$

where $W_{T|T}^{i} = W_{T}^{i}$ then by using (7) and (6), we obtain the approximation

$$\widehat{p}_{\theta}\left(dx_{n} \middle| y_{0:T} \right) = \sum_{i=1}^{N} W_{n|T}^{i} \delta_{X_{n}^{i}}(dx_{n})$$

with

$$W_{n|T}^{i} = W_{n}^{i} \times \sum_{j=1}^{N} \frac{W_{n+1|T}^{j} f_{\theta}\left(X_{n+1}^{j} | X_{n}^{i} \right)}{\sum_{l=1}^{N} W_{n}^{l} f_{\theta}\left(X_{n+1}^{j} | X_{n}^{l} \right)}. \tag{8}$$

Particle Implementation

- Forward Filtering Backward Smoothing (FFBSm) :

  - Run a particle filter from time $n = 0$ to T, storing the approximate filtering distributions $\{\widehat{p}_\theta\left(dx_n \middle| y_{0:n}\right)\}_{n=0}^{T}$,
  - Initialise backward pass: $W^i_{T|T} = W^i_T$
  - for $n = T-1, T-2, \ldots, 0$ compute weights

$$W^i_{n|T} = W^i_n \times \sum_{j=1}^{N} \frac{W^j_{n+1|T} f_\theta\left(X^j_{n+1} | X^i_n\right)}{\sum_{l=1}^{N} W^l_n f_\theta\left(X^j_{n+1} | X^l_n\right)}. \tag{9}$$

  and obtain the approximation

$$\widehat{p}_\theta\left(dx_n \middle| y_{0:T}\right) = \sum_{i=1}^{N} W^i_{n|T} \delta_{X^i_n}(dx_n)$$

Particle Implementation

- Lets say we have performed Forward Filtering Backward Smoothing (FFBSm) :
- Assume we have an approximation

$$\widehat{p}_\theta\left(dx_{n+1} \middle| y_{0:T}\right) = \sum_{i=1}^{N} W^i_{n+1|T}\delta_{X^i_{n+1}}(dx_{n+1})$$

and are interested to obtain the approximation

$$\widehat{p}_\theta\left(dx_n, dx_{n+1} \middle| y_{0:T}\right) = \sum_{i=1}^{N} \tilde{W}^i_{n,n+1|T}\delta_{X_n^{a_{n+1}(i)}, X^i_{n+1}}(dx_n)$$

with $X_n^{a_{n+1}(i)}$ being the ancestor of X^i_{n+1} then we can weight the pair $X_n^{a_{n+1}(i)}, X^i_{n+1}$ by

$$\tilde{W}^i_{n,n+1|T} = W_n^{a_{n+1}(i)} \frac{W^i_{n+1|T} f_\theta\left(X^i_{n+1}|X_n^{a_{n+1}(i)}\right)}{\sum_{l=1}^{N} W^l_n f_\theta\left(X^i_{n+1}|X^l_n\right)}. \qquad (10)$$

Discussion

- Both FFBSa and FFBSm have computational cost is prop. to $N^2 T$ operations in total
- Assuming exponential forgetting of HMM:
  - $\mathcal{S}_n^\theta$ based on the fixed-lag approximation has an asymptotic variance with rate n/N with a non-vanishing (as $N \to \infty$) bias proportional to n and a constant decreasing exponentially fast with L.
  - The asymptotic bias and variance of the particle estimate of $\mathcal{S}_n^\theta$ computed using FFBSa/m satisfy:

$$\left| \mathbb{E}\left(\widehat{\mathcal{S}}_n^\theta\right) - \mathcal{S}_n^\theta \right| \leq F_\theta \frac{n}{N}, \quad \mathbb{V}\left(\widehat{\mathcal{S}}_n^\theta\right) \leq H_\theta \frac{n}{N}. \tag{11}$$

Discussion

- To compute $\widehat{\mathcal{S}}_n^\theta$ one can implement with cost $N^2 T$
  1. simple particle filter with N^2 particles
  2. FFBS particle filter with N particles
- Then
  - Case 1: suffers from path degeneracy
    - bias of order T/N^2
    - variance at least of order T^2/N^2
  - Case 2: more expensive
    - bias of order T/N
    - variance of order T/N

Numerical example

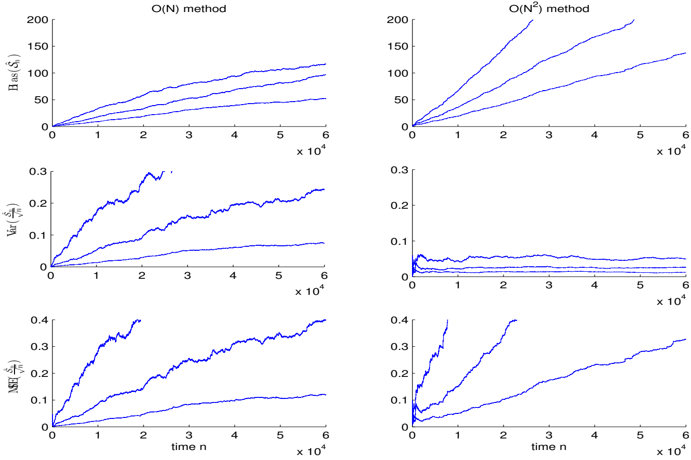

Figure: Estimating smoothed additive functionals: Empirical bias of the estimate of $\mathcal{S}_n^\theta$ (top panel), empirical variance (middle panel) and mean squared error (bottom panel) for the estimate of $\mathcal{S}_n^\theta/\sqrt{n}$.

On-line methods

- On-line/ Forwards only extensions for smoothing methods exist
  - Poyiadjis, Doucet, Singh 11 Particle approximations of the score and observed information matrix...
  - Del Moral, Doucet, Singh 09 Forward Smoothing using Sequential Monte Carlo
  - Olsson and Westerborn 17 Efficient particle-based online smoothing in general hidden Markov models: The PaRIS algorithm

Discussion

- Particle smoothing is necessary for approximation of $p_\theta(x_n|y_{0:T})$
- For smoothed additive functionals $\mathcal{S}_n^\theta$ when $\mathcal{O}(N^2 T)$ cost is used
  - FFBSm/Sa will have very good variance properties
  - simple PF with N^2 particles will have better bias properties
- PaRiS can reduce computational cost of particle smoothing
- Forward smoothing methods are basis for online parameter inference
- Tutorial on filtering and smoothing: `http://www.stats.ox.ac.uk/~doucet/doucet_johansen_tutorialPF2011.pdf`