Sequential Monte Carlo smoother for partially observed diffusion processes

M.P. Etienne, P. Gloaguen, S. Le Corff

Masterclass in Bayesian Statistics at CIRM

Motivating example: Estimating fish biomass

What we know

Scientific observations Y

What we want

Population biomass X

Biomass dynamic models observed through abundance indices

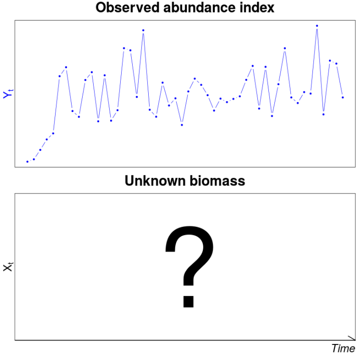
Motivating example: Model based approach

$$dX_t = \kappa X_t \left(1 - \frac{X_t}{\gamma}\right) dt + \sigma X_t dW_t, \quad X_0 \sim \chi_0 \text{ s.t. } X_0 \overset{a.s}{>} 0 \quad \text{Diffusion process}$$

$$Y_k = cX_{t_k} \exp(\varepsilon_k), \quad \varepsilon_k \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2_{obs}), \; k = 0, \dots, n \qquad \text{Observations}$$

Population dynamics model

- X_t: True biomass at year t;
- Y_k: Abundance index at year t_k;
- γ: Carrying capacity;
- κ: Recruitement rate;
- $\sigma, \sigma_{\text{obs}}$: Innovation and error parameters;
- c: Detectability;
- $t_0, \dots, t_n$ are the $(n+1)$ observation times.



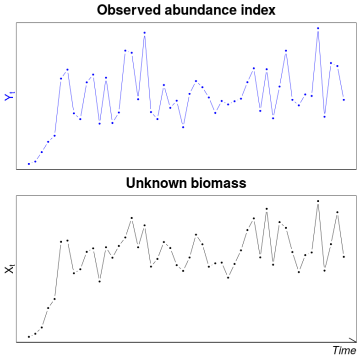
Motivating example: Model based approach

$$dX_t = \kappa X_t \left(1 - \frac{X_t}{\gamma}\right) dt + \sigma X_t dW_t, \quad X_0 \sim \chi_0 \text{ s.t. } X_0 \overset{a.s}{>} 0 \quad \text{Diffusion process}$$

$$Y_k = c X_{t_k} \exp(\varepsilon_k), \quad \varepsilon_k \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2_{obs}), \; k = 0, \dots, n \qquad \text{Observations}$$

Classical Inference problems

1. From abundance index, **estimate the true biomass**.
2. From abundance index, **estimate model parameters**

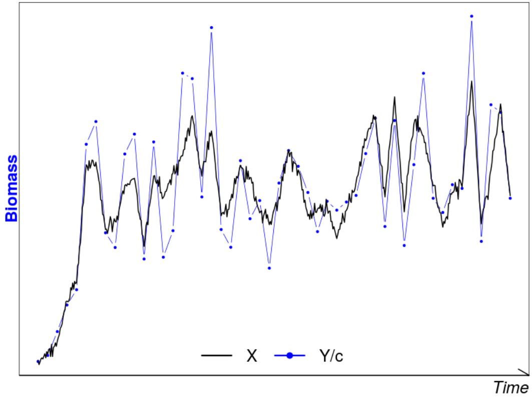


Motivating example

Population dynamics model

$$dX_t = \kappa X_t \left(1 - \frac{X_t}{\gamma}\right) dt + \sigma X_t dW_t, \quad X_0 = x_0 > 0, \qquad \text{Markov process}$$

$$Y_k = c X_k \exp(\varepsilon_k), \quad \varepsilon_k \overset{i.i.d.}{\sim} \mathcal{N}(\mu_{obs}, \sigma^2_{obs}), \qquad \text{Observations}$$



Context: General State Space Model

$$dX_t = \kappa X_t \left(1 - \frac{X_t}{\gamma}\right) dt + \sigma X_t dW_t, \quad X_0 \sim \chi_0 \text{ s.t. } X_0 \overset{a.s}{>} 0 \quad \text{Diffusion process}$$

$$Y_k = cX_{t_k} \exp(\varepsilon_k), \quad \varepsilon_k \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2_{obs}), \quad \text{Observations}$$

Context: General State Space Model

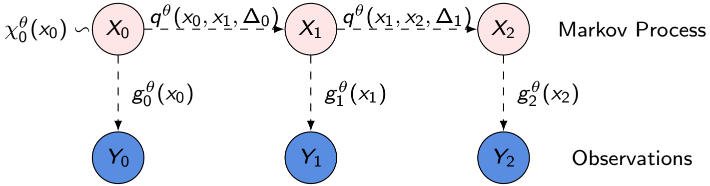
$$\mathrm{d}X_t = \alpha^\theta(X_t)\mathrm{d}t + \beta^\theta(X_t)\mathrm{d}W_t, \quad X_0 \sim \chi_0^\theta \qquad \text{Diffusion process}$$

$$Y_k \sim g_k^\theta(X_{t_k}, \theta) \qquad \text{Observations}$$

Partially observed diffusion process

- $(X_t)_{t\geq 0}$ is solution to a stochastic differential equation (SDE);
- The solution to a SDE is a continuous time Markov process;
- $(Y_k)_{k=0,\dots,n}$ are independent conditionally to $(X_t)_{t\geq 0}$;
- The model therefore defines a specific Hidden Markov Model;
- Model known as partially observed diffusion process;

Context: General State Space Model



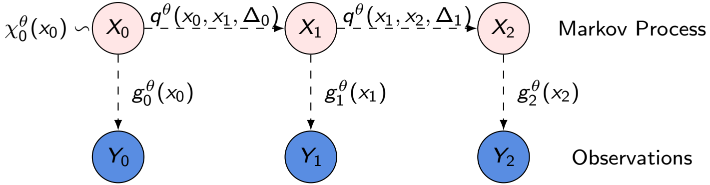
- Observations $(Y_k)_{k=0,\dots,n}$ at times $t_0 = 0, t_1 = t_0 + \Delta_0, \dots, t_n = t_{n-1} + \Delta_{n-1}$.
- Unobserved Markov process $(X_t)_{0 \le t \le t_n}$ (**notation**: $X_k := X_{t_k}$);
- $X_t \in \mathbb{R}^d$, $Y_k \in \mathbb{R}^{d'}$;
- A set of parameters θ;

$$X_0 \sim \chi_0^\theta(x_0) \qquad \text{Initial density}$$

$$X_k | (X_{k-1} = x_{k-1}) \sim q^\theta(x_{k-1}, x_k, \Delta_{k-1}) \qquad \text{Transition density}$$

$$Y_k | (X_k = x_k) \sim g^\theta(x_k, y_k) := g_k^\theta(x_k) \qquad \text{Observation density}$$

Context: Partially observed diffusion processes



$(X_t)_{0 \leq t \leq t_n}$ is supposed to be solution to the stochastic differential equation:

$$\mathrm{d}X_t = \alpha^\theta(X_t)\mathrm{d}t + \beta^\theta(X_t)\mathrm{d}W_t, \quad X_0 \sim \chi_0^\theta(x_0) \tag{1}$$

Context: Partially observed diffusion processes

Particularity of SDE based models

In general,

- the transition density $q^{\theta}(\cdot)$ is not explicit except in very specific cases (even when θ is known).
- Exact simulation is not straightforward (except in the same specific cases).

Context: Partially observed diffusion processes

SDE based models in practice

In general,

- Approximation of $q^{\theta}(\cdot)$ (Hermite expansion for example)
- Simulation using EA algorithm (Beskos et al., 2006), or moslty approximated numerical scheme (Euler, Ozaki, ...).

Objective

Examples of application

- Parameter estimation : E step of EM algorithm:

$$\mathcal{Q}(\theta_0, \theta) = \mathbb{E}[\ell(\theta; X_{0:n}, Y_{0:n}) | Y_{0:n}; \theta_0]$$

- Path reconstruction, target tracking:

$$\mathbb{E}[X_k | Y_{0:n}; \theta]$$

General goal

Computing:

$$\mathbb{E}\left[H_n(X_{0:n}) | Y_{0:n}; \theta\right]$$

Definitions and notations

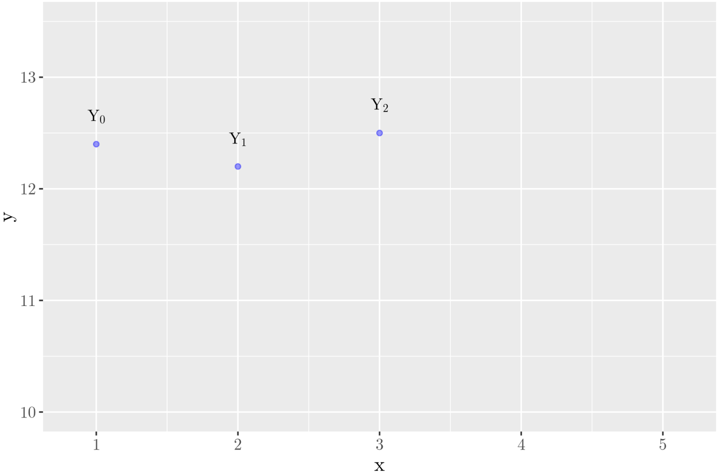
Goal

Compute

$$\mathbb{E}\left[H_n(X_{0:n})\middle| Y_{0:n};\theta\right]$$

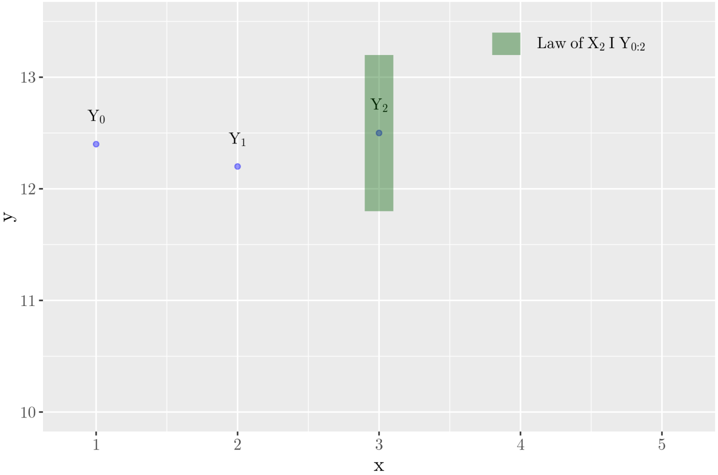
- Filtering distribution, the law of $X_k|Y_{0:k}$ (observations until time k)
  - Expectation: $\phi_k[f] = \mathbb{E}\left[f(X_k)\middle|Y_{0:k}\right]$.
- Smoothing distribution, the law of $X_{k:\ell}|Y_{0:n}$ (all observations)
  - Expectation: $\phi_{k:\ell|n}[f] = \mathbb{E}\left[f(X_{k:\ell})\middle|Y_{0:n}\right]$.

Difference between filtering and smoothing, where is X_2?



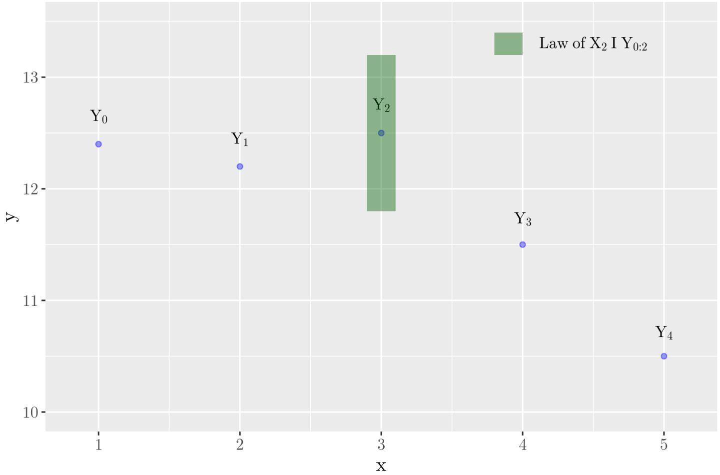
Filtering distribution, law of $X_2 | Y_{0:2}$

Difference between filtering and smoothing, where is X_2?



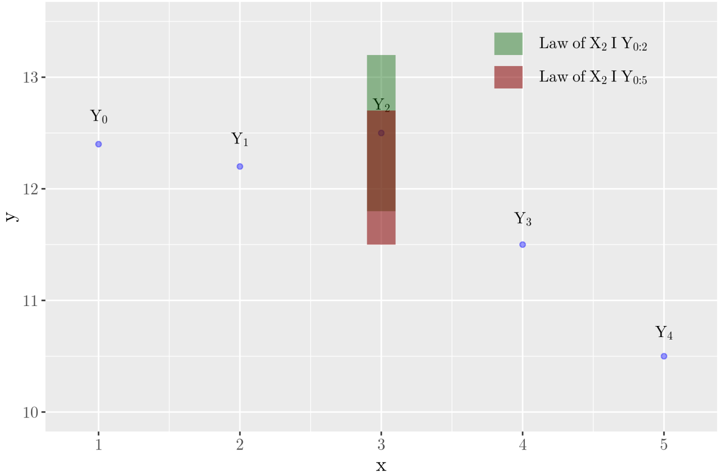
Filtering distribution, law of $X_2|Y_{0:2}$

Difference between filtering and smoothing, where is X_2?



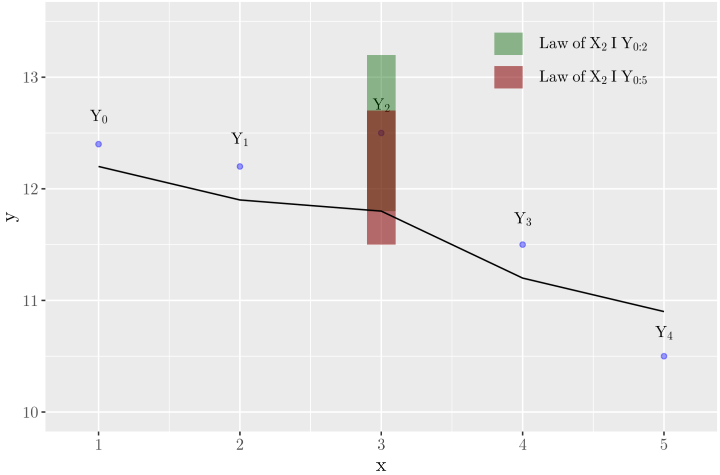
Filtering distribution, law of $X_2 | Y_{0:2}$

Difference between filtering and smoothing, where is X_2?



Smoothing distribution, law of $X_2 | Y_{0:4}$, takes all observations into account!

Difference between filtering and smoothing, where is X_2?



Smoothing distribution, law of $X_2|Y_{0:4}$, takes all observations into account!

Definitions and notations

Goal

Compute

$$\mathbb{E}\left[H_n(X_{0:n})|Y_{0:n};\theta\right] = \phi_{0:n|n}[H_n]$$

- Filtering distribution, the law of $X_k|Y_{0:k}$ (observations until time k)
  - Expectation: $\phi_k[f] = \mathbb{E}\left[f(X_k)|Y_{0:k}\right]$.
- Smoothing distribution, the law of $X_{k:\ell}|Y_{0:n}$ (all observations)
  - Expectation: $\phi_{k:\ell|n}[f] = \mathbb{E}\left[f(X_{k:\ell})|Y_{0:n}\right]$.

Unknown in general!

$\rightarrow$ Approximation using **Sequential Monte Carlo** methods:
Approximation of ϕ_k (resp. $\phi_{k:\ell|n}$) by ϕ_k^N (resp. $\phi_{k:\ell|n}^N$) such that for all function f:

$$\phi_k^N[f] \approx \phi_k[f]$$

$$\text{Resp. } \phi_{k:\ell|n}^N[f] \approx \phi_{k:\ell|n}[f]$$

Previous work and contribution

Goal

Compute

$$\mathbb{E}\left[H_n(X_{0:n})\middle|Y_{0:n};\theta\right] = \phi_{0:n|n}[H_n]$$

SSM with known transition density $q^\theta(x_t, x_{t+\Delta}, \Delta)$

- First particle filter: Gordon et al. (1993);
- Backward particle smoother with linear complexity: Douc et al. (2011);
- Online* particle smoother: Olsson et al. (2017).

Partially observed diffusion processes (unknown $q^\theta(x_t, x_{t+\Delta}, \Delta)$)

- First unbiased particle filter: Fearnhead et al. (2008);
- Biased smoother: Olsson and Strojby (2011);
- Online* unbiased smoother: Gloaguen et al. (2018).

* For functionals H_n s.t. $H_n(X_{0:n}) = \sum_{k=0}^{n-1} h_k(X_k, X_{k+1})$.

Particle filtering for SSM

Idea

For a function f, approximating the expectation $\phi_k[f] = \mathbb{E}\left[f(X_k) | Y_{0:k}\right]$ by a finite sum ϕ_k^N:

$$\phi_k^N[f] = \sum_{i=1}^{N} \omega_k^i f(\xi_k^i)$$

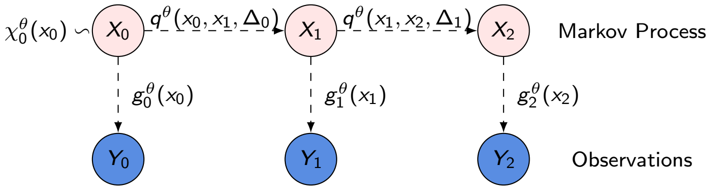
where

- $(\xi_k^i)_{i=1,\dots,N}$ is a finite set of N **particles**;
- $(\omega_k^i)_{i=1,\dots,N}$ are the respective **importance weights** of the N particles;
- $\sum_{i=1}^{N} \omega_k^i = 1$

Asymptotic property

$$\phi_k^N[f] \underset{N \to \infty}{\longrightarrow} \phi_k[f]$$

Particle filtering for SSM



Filtering problem: Approximating the distribution of $X_k|Y_{0:k}$

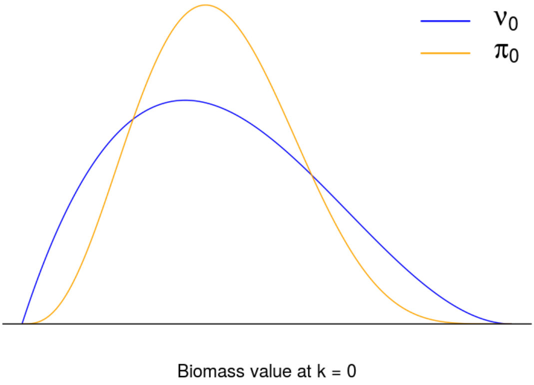
First: Let's suppose that $q^\theta(x, y, \Delta)$ can be computed (for a given θ).

Particle filtering for SSM: Initialization, $k = 0$

Approximation of the law of $X_0|Y_0$.

- **True distribution** $\pi_0^\theta(\cdot)$, law of $X_0|Y_0$;
- **Proposition distribution** $p_0^\theta(\cdot)$

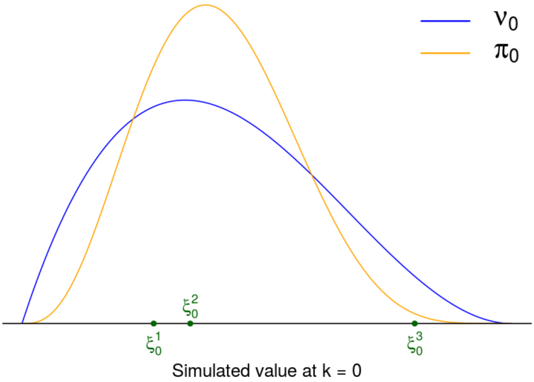
Choosing distribution



Particle filtering for SSM: Initialization, $k = 0$

Approximation of the law of $X_0|Y_0$.

- **True distribution** $\pi_0^\theta(\cdot)$, law of $X_0|Y_0$;
- **Proposition distribution** $p_0^\theta(\cdot)$



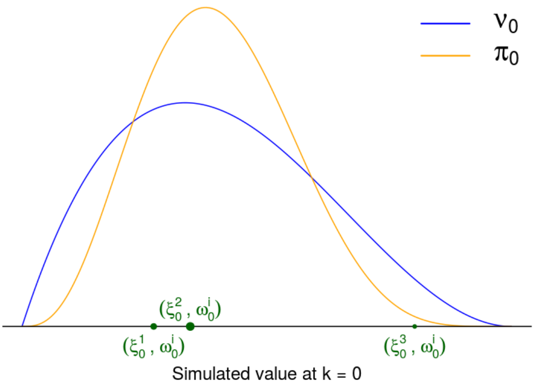
Sampling

1. Sample $\xi_0^1, \ldots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$

Particle filtering for SSM: Initialization, $k = 0$

Approximation of the law of $X_0|Y_0$.

- **True distribution** $\pi_0^\theta(\cdot)$, law of $X_0|Y_0$;
- **Proposition distribution** $p_0^\theta(\cdot)$



Weighting

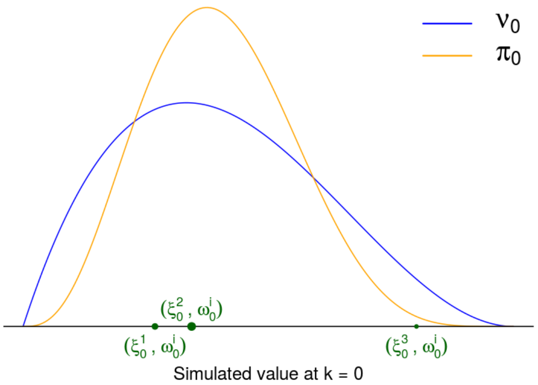
1. Sample $\xi_0^1, \ldots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{\pi_0^\theta(\xi_0^i|Y_0)}{p_0^\theta(\xi_0^i)}$$

Particle filtering for SSM: Initialization, $k = 0$

Approximation of the law of $X_0|Y_0$.

- **True distribution** $\pi_0^\theta(\cdot)$, law of $X_0|Y_0$;
- **Proposition distribution** $p_0^\theta(\cdot)$



Weighting

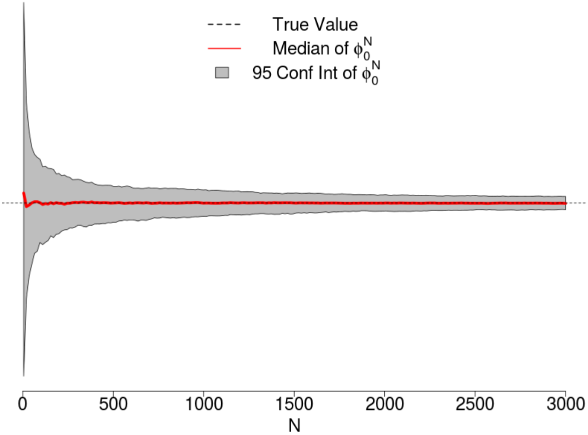
1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

Particle filtering for SSM: Initialization, $k = 0$

Approximation of the law of $X_0 | Y_0$.

- **True distribution** $\pi_0^\theta(\cdot)$, law of $X_0 | Y_0$;
- **Proposition distribution** $p_0^\theta(\cdot)$



Weighting

1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

3. Approximate $\mathbb{E}[X_0 | Y_0]$:

$$\mathbb{E}[X_0 | Y_0] \simeq \sum_{i=1}^{N} \omega_0^i f(\xi_0^i)$$

Particle filtering for SSM, propagation to $k > 0$

Approximating the law of $X_k|\mathbf{Y_{0:k}},\ k > 0$

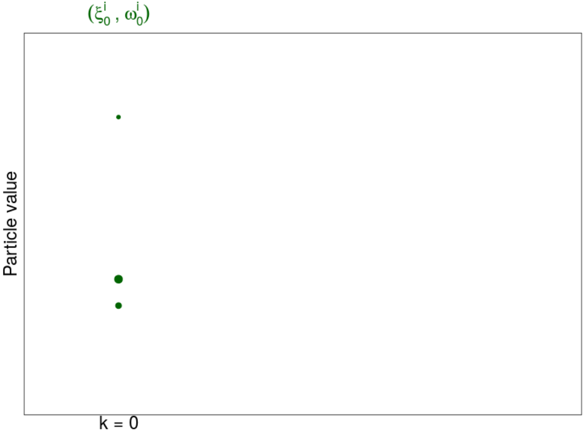
Particle based method

- Propagate simulated particles at time 0 to create new particles at time 1;
- Compute importance weights for these new particles;
- Propagate these new particles at time 2, compute weights and so on...

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k|\mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Initialization

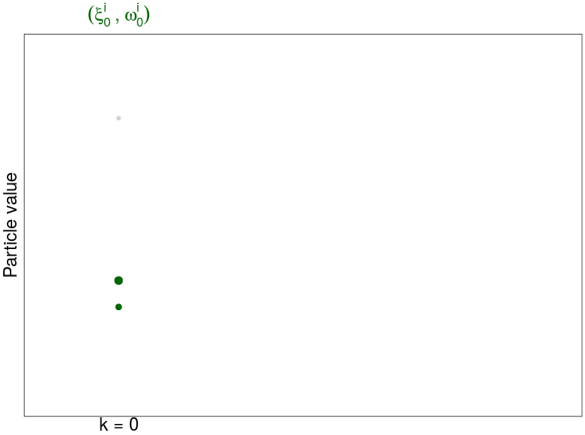
1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k | \mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Selection

1. Sample $\xi_0^1, \ldots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

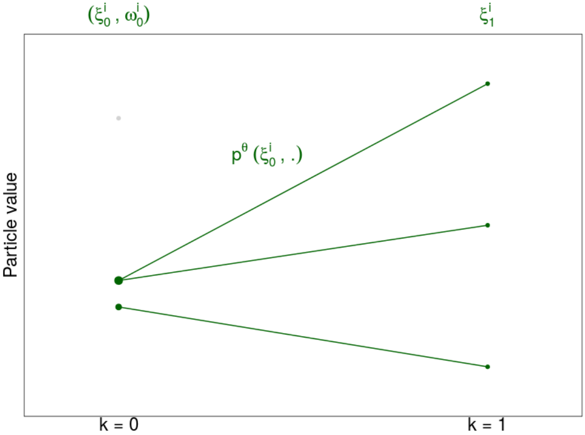
$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

3. Sample $I_0^i \propto \{\omega_0^i\}_{i=1,\ldots,N}$

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k | \mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Propagation

1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

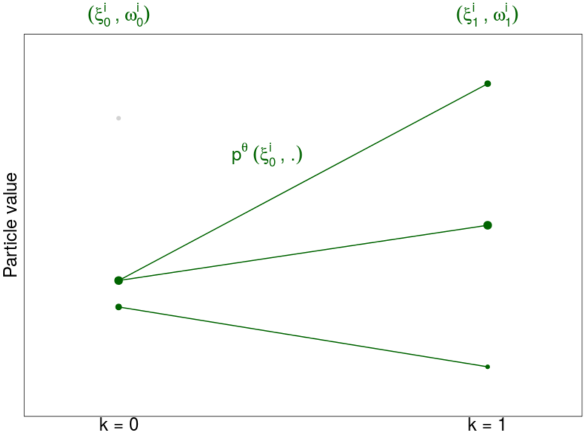
$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

3. Sample $I_0^i \propto \{\omega_0^i\}_{i=1,\dots,N}$
4. Sample $\xi_1^i \sim p^\theta(\xi_0^{I_0^i}, \cdot)$

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k | \mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Weighting

1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

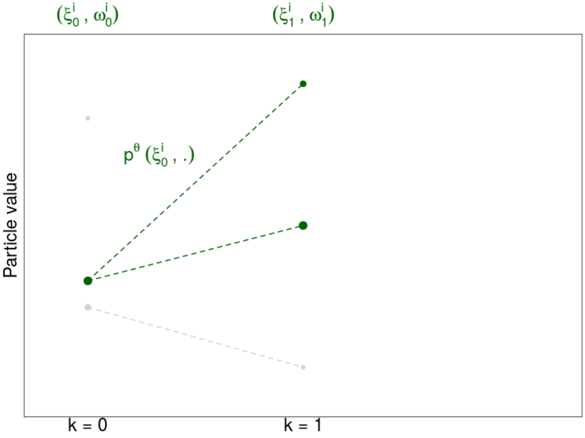
3. Sample $I_0^i \propto \{\omega_0^i\}_{i=1,\dots,N}$
4. Sample $\xi_1^i \sim p^\theta(\xi_0^{I_0^i}, \cdot)$
5. Compute

$$\omega_1^i = \frac{g_1^\theta(\xi_1^i) q^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}{p^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}$$

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k|\mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Selection

1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

3. Sample $I_0^i \propto \{\omega_0^i\}_{i=1,\dots,N}$
4. Sample $\xi_1^i \sim p^\theta(\xi_0^{I_0^i}, \cdot)$
5. Compute

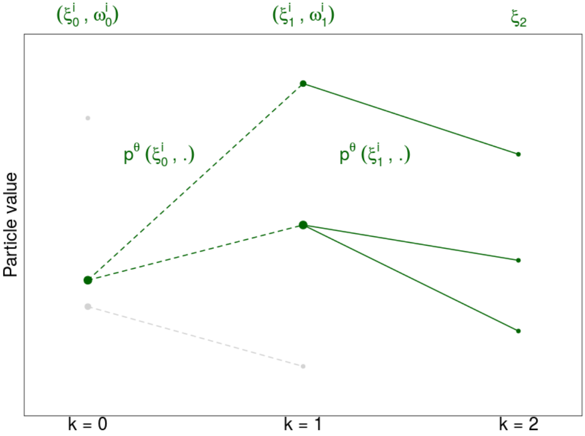
$$\omega_1^i = \frac{g_1^\theta(\xi_1^i)q^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}{p^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}$$

And so on...

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k|\mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Propagation

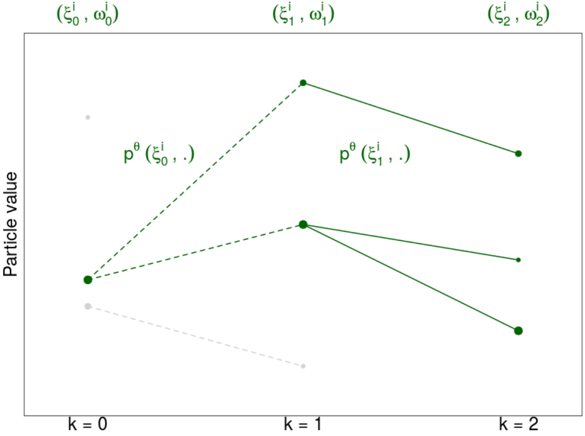
1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)
$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$
3. Sample $I_0^i \propto \{\omega_0^i\}_{i=1,\dots,N}$
4. Sample $\xi_1^i \sim p^\theta(\xi_0^{I_0^i}, \cdot)$
5. Compute
$$\omega_1^i = \frac{g_1^\theta(\xi_1^i)q^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}{p^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}$$

And so on...

Particle filtering for SSM, propagation to $k > 0$

Approximating the law $X_k | \mathbf{Y_{0:k}}$.

- Requires as proposition law a propagation distribution $p^\theta(x, \cdot)$.



Weighting

1. Sample $\xi_0^1, \dots \xi_0^N \overset{i.i.d.}{\sim} p_0^\theta(\cdot)$
2. Compute (and normalize)

$$\omega_0^i = \frac{g_0^\theta(\xi_0^i)\chi_0^\theta(\xi_0^i)}{p_0^\theta(\xi_0^i)}$$

3. Sample $I_0^i \propto \{\omega_0^i\}_{i=1,\dots,N}$
4. Sample $\xi_1^i \sim p^\theta(\xi_0^{I_0^i}, \cdot)$
5. Compute

$$\omega_1^i = \frac{g_1^\theta(\xi_1^i) q^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}{p^\theta(\xi_0^{I_0^i}, \xi_1^i, \Delta_0)}$$

And so on...

Particle filtering for POD processes

Problem

- In POD processes, $q^\theta(x, y, \Delta)$ can't be computed (even when θ is known);
- This quantity is crucial for weights computation.

General Poisson estimator, Fearnhead et al. (2008)

Under some assumptions, there exists an unbiased estimator $\hat{q}^\theta(x_k, x_{k+1}, \Delta_k, \zeta_k)$ such that

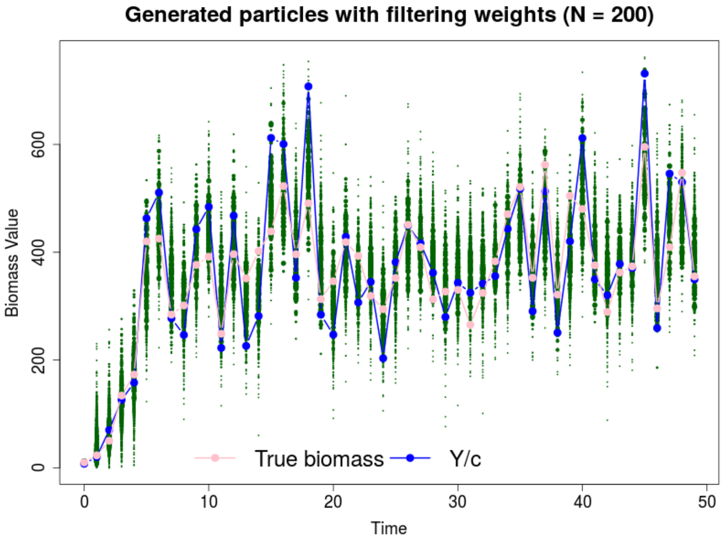
$$\hat{q}^\theta(x_k, x_{k+1}, \Delta_k, \zeta_k) > 0 \text{ and } \mathbb{E}[\hat{q}^\theta(x_k, x_{k+1}, \Delta_k, \zeta_k)] = q^\theta(x_k, x_{k+1}, \Delta_k)$$

ζ_k is a random variable requiring simulation of constrained Brownian bridges.

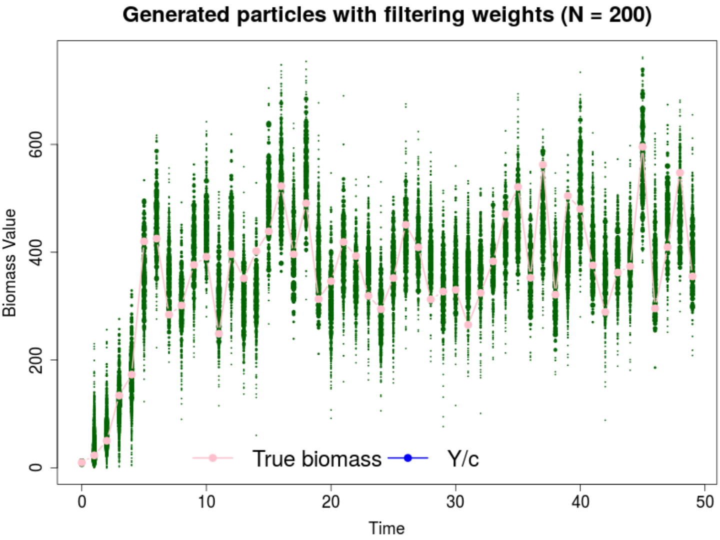
New filtering weights

$\omega_k^i = \frac{g_1^\theta(\xi_k^i) q^\theta(\xi_{k-1}^{I_k^i}, \xi_k^i, \Delta_{k-1})}{p^\theta(\xi_{k-1}^{I_{k-1}^i}, \xi_k^i, \Delta_{k-1})}$ is replaced by $\hat{\omega}_k^i = \frac{g_1^\theta(\xi_k^i) \hat{q}^\theta(\xi_{k-1}^{I_k^i}, \xi_k^i, \Delta_{k-1}, \zeta_{k-1})}{p^\theta(\xi_{k-1}^{I_{k-1}^i}, \xi_k^i, \Delta_{k-1})}$

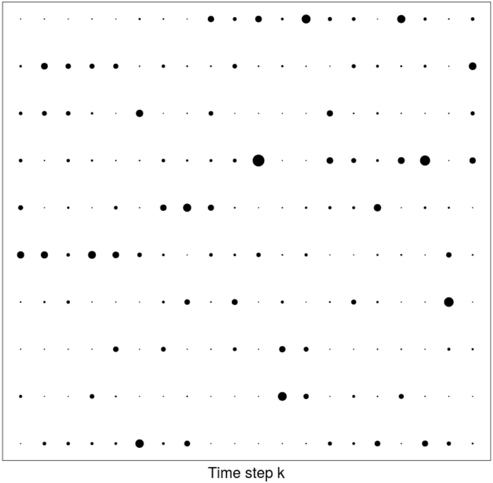
Particle filtering for SSM, back to our example



Particle filtering for SSM, back to our example



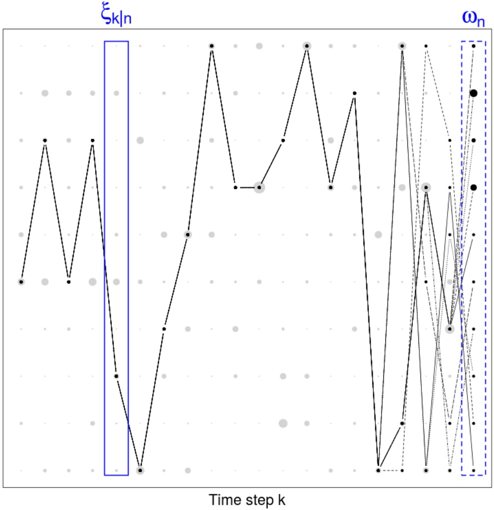
Particle smoothing, naïve and direct approach



Approx. $\phi_{k|n}$ (law of $X_k | Y_{0:n}$)

- Naïve approach
  - Run a PF until n;
  - Remember ancestors of each particle;

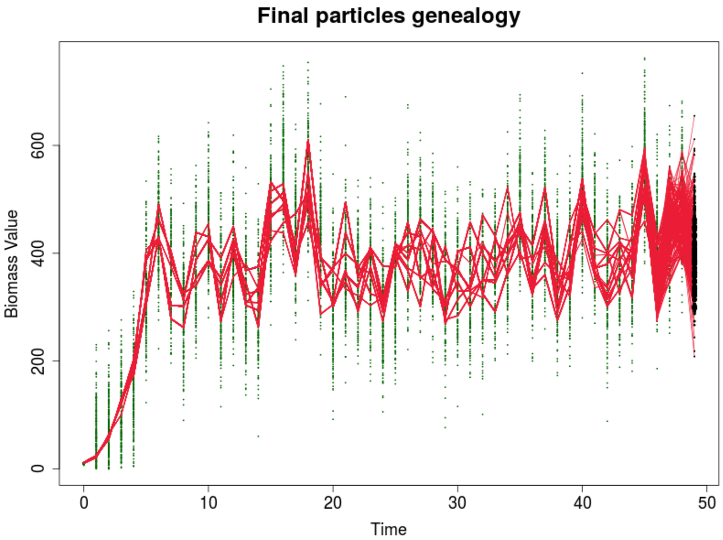
Particle smoothing, naïve and direct approach



Approx. $\phi_{k|n}$ (law of $X_k | Y_{0:n}$)

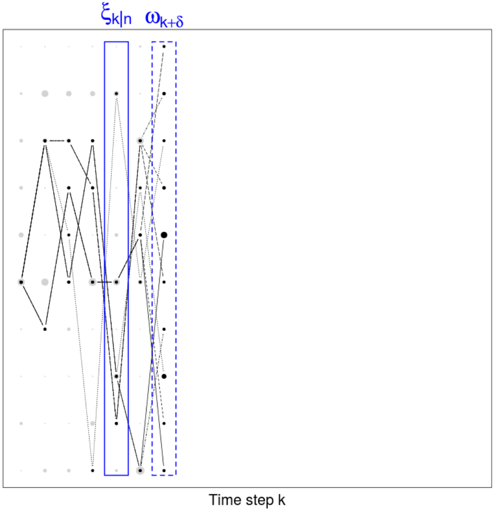
- Naïve approach
  - Run a PF until n;
  - Remember ancestors of each particle;
  - Approx. $\phi_{k|n}$ with ancestors of ξ_n^i, at generation k having weights ω_n^i;
  - High variance!

Particle smoothing, the genealogy our example



Early biomasses distributions are approximated with very few particles!

Particle smoothing, fixed lag technique



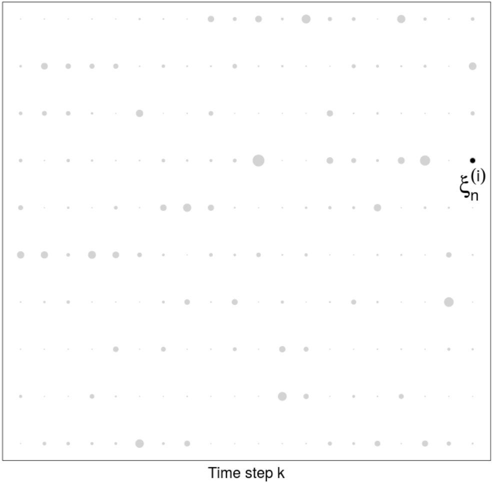
Approx. $\phi_{k|n}$ (law of $X_k|Y_{0:n}$)

- Fixed lag technique
  - Choose a lag δ;
  - Run a PF until $k+\delta$
  - Approx. $\phi_{k|n}$ by $\phi_{k|k+\delta}$ with ancestors of $\xi^i_{k+\delta}$ having weights $\omega^i_{k+\delta}$
  - Variance $\uparrow$ when $\delta\uparrow$
  - Biased technique, with bias $\uparrow$ when $\delta\downarrow$

Used by Olsson and Strojby (2011), first (biased) smoother for PODs

Particle smoothing, backward simulation

Idea: Reducing the variance by recreating diversity

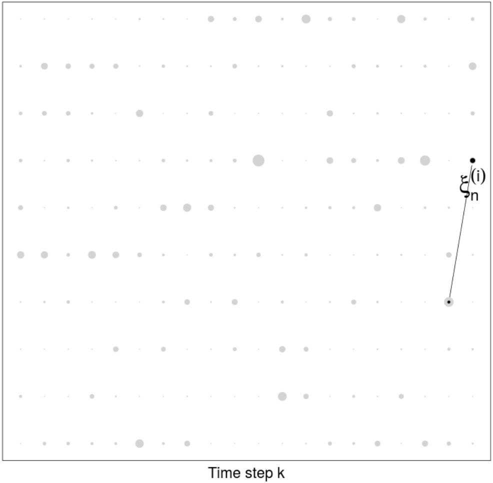


Approx. $\phi_{k|n}$ (law of $X_k | Y_{0:n}$)

- Backward simulation
  - Run a PF until n;

Particle smoothing, backward simulation

Idea: Reducing the variance by recreating diversity

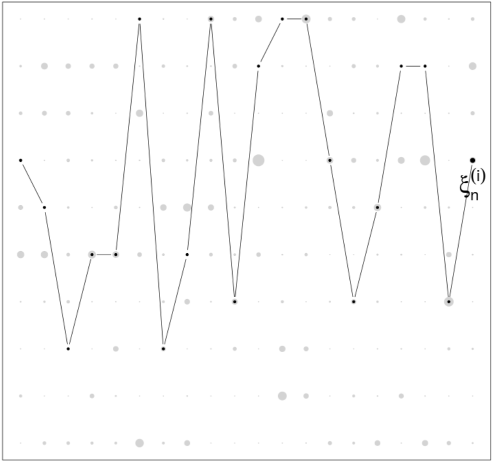


Approx. $\phi_{k|n}$ (law of $X_k|Y_{0:n}$)

- Backward simulation
  - Run a PF until n;
  - For a final particle ξ_n^i, pick a "probable" direct ancestor;

Particle smoothing, backward simulation

Idea: Reducing the variance by recreating diversity



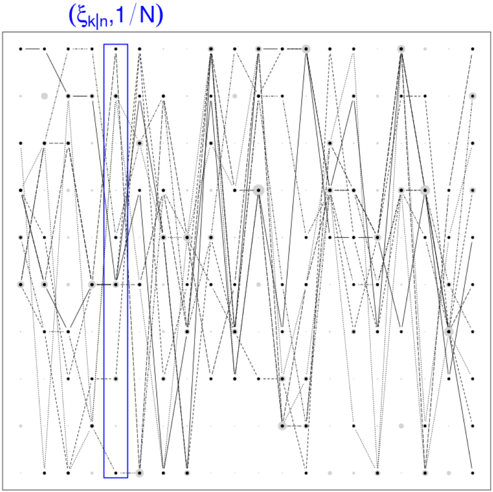
Time step k

Approx. $\phi_{k|n}$ (law of $X_k | Y_{0:n}$)

- Backward simulation
  - Run a PF until n;
  - For a final particle ξ_n^i, pick a "probable" direct ancestor;
  - Repeat until reaching an initial particle;

Particle smoothing, backward simulation

Idea: Reducing the variance by recreating diversity

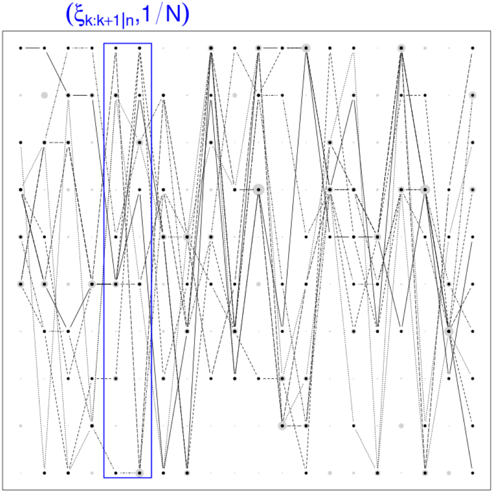


Approx. $\phi_{k|n}$ (law of $X_k|Y_{0:n}$)

- Backward simulation
  - Run a PF until n;
  - For a final particle ξ_n^i, pick a "probable" direct ancestor;
  - Repeat until reaching an initial particle;
  - Do it for each final point;
  - Approximate $\phi_{k|n}$ with particles obtained at time k, having new weights $1/N$;

Particle smoothing, backward simulation

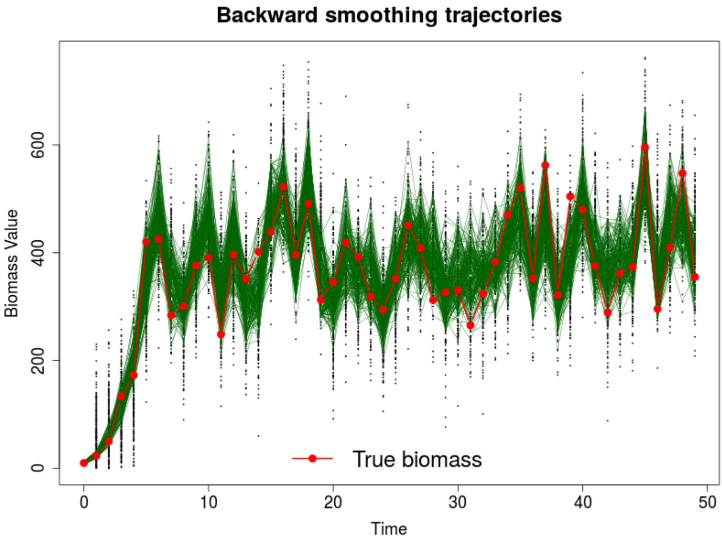
Idea: Reducing the variance by recreating diversity



Approx. $\phi_{k|n}$ (law of $X_k | Y_{0:n}$)

- Backward simulation
  - Run a PF until n;
  - For a final particle ξ_n^i, pick a "probable" direct ancestor;
  - Repeat until reaching an initial particle;
  - Do it for each final point;
  - Approximate $\phi_{k|n}$ with particles obtained at time k, having new weights $1/N$;
  - Possibility of having joint distributions.

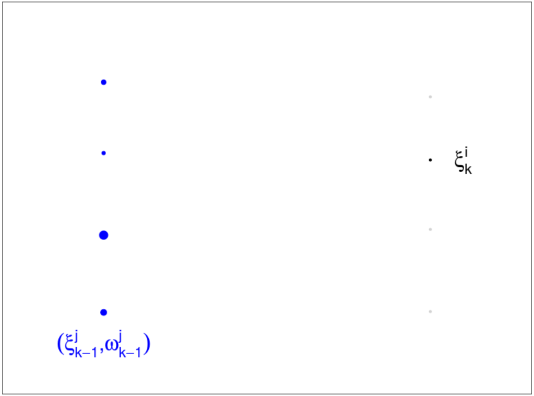
Backward sampling on our example



How can it be done when $q^{\theta}(x, y, \Delta)$ can't be computed?

Backward sampling mechanism

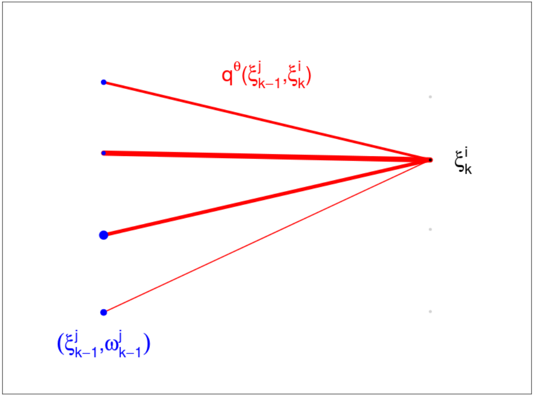
Sampling an ancestor of ξ_k^i.



How credible is each ξ_{k-1}^j as the parent of ξ_k^i?

Backward sampling mechanism

Sampling an ancestor of ξ_k^i.
Draw ancestor with prob. $\Lambda_{k-1}^{j,i}$, i.e.: **"The probability of ξ_{k-1}^j being the ancestor of ξ_k^i under the true dynamics".**



How credible is each ξ_{k-1}^j as the parent of ξ_k^i?

$$\Lambda_{k-1}^{j,i} \propto \underbrace{\hat{\omega}_{k-1}^j}^{\text{Ancestor's filtering weight}} \times \underbrace{q^\theta(\xi_{k-1}^j, \xi_k^i, \Delta_{k-1})}_{\text{Markov transition density}}$$

Sampling ancestor with weights $\Lambda_{k-1}^{j,i}$ in the POD context

Main problem

Backward smoothing requires to sample an ancestor (value of j) with weights:

$$\Lambda_{k-1}^{j,i} = \frac{\hat{\omega}_{k-1}^{j} q^{\theta}(\xi_{k-1}^{j}, \xi_{k}^{i}, \Delta_{k-1})}{\sum_{\ell=1}^{N} \hat{\omega}_{k-1}^{\ell} q^{\theta}(\xi_{k-1}^{\ell}, \xi_{k}^{i}, \Delta_{k-1})}$$

The probability of ξ_{k-1}^{j} being the ancestor of ξ_{k}^{i} under the true dynamics

The solution? (When q^{θ} can't be computed)

Use the unbiased estimator $\hat{q}^{\theta}(\xi_{k-1}^{j}, \xi_{k}^{i}, \Delta_{k-1}, \zeta_i)$ as a substitute:

$$\hat{\Lambda}_{k-1}^{j,i} = \frac{\hat{\omega}_{k-1}^{j} \hat{q}^{\theta}(\xi_{k-1}^{j}, \xi_{k}^{i}, \Delta_{k-1}, \zeta_i)}{\sum_{\ell=1}^{N} \hat{\omega}_{k-1}^{\ell} \hat{q}^{\theta}(\xi_{k-1}^{j}, \xi_{k}^{i}, \Delta_{k-1}, \zeta_i)}$$

But $\mathbb{E}(\hat{\Lambda}_{k-1}^{j,i}) \neq \Lambda_{k-1}^{j,i}$, because of the ratio.

Sampling with weights $\Lambda_{k-1}^{j,i}$ in the POD context

$$\Lambda_{k-1}^{j,i} = \frac{\hat{\omega}_{k-1}^{j} q^{\theta}(\xi_{k-1}^{j}, \xi_k^i, \Delta_{k-1})}{\sum_{\ell=1}^{N} \hat{\omega}_{k-1}^{\ell} q^{\theta}(\xi_{k-1}^{\ell}, \xi_k^i, \Delta_{k-1})}$$

Lemma Gloaguen et al. (2018)

Assumption: $\exists\ \hat{\sigma}_{k,+}$ such that $\forall x, y,\ 0 < \hat{q}^{\theta}(x, y, \Delta_k, \zeta_k) < \hat{\sigma}_{k,+}$ a.s.
Consider the random variable $\widehat{\mathrm{J}}$ defined as follow:

- Sample $J_{cand} \in 1, \ldots, N$ with probabilities $\propto \{\hat{\omega}_{k-1}^{i}\}_{i=1,\ldots,N}$;
- Sample ζ_{k-1} using the GPE of Fearnhead et al. (2008);
- Sample $U \sim \mathcal{U}[0,1]$;

if $U \leq \frac{\hat{q}^{\theta}(\xi_{k-1}^{J_{cand}}, \xi_k^i, \Delta_{k-1}, \zeta_{k-1})}{\hat{\sigma}_{k-1,+}}$ **then**
    Set $\widehat{\mathrm{J}} = J_{cand}$;
else
    Try Again;
end if

Then,

$$\mathbb{P}(\widehat{\mathrm{J}} = j) = \Lambda_{k-1}^{j,i}$$

Smoothing for POD processes

Proposition: Asymptotic unbiased property Gloaguen et al. (2018)

Let denote $\hat{\phi}_{0:n|n}[H_n]$ our estimator of $\phi_{0:n|n}[H_n] := \mathbb{E}[H_n(X_{0:n})]$:

$$\mathbb{P}\left(\left|\hat{\phi}_{0:n|n}[H_n] - \phi_{0:n|n}[H_n]\right| \geq \varepsilon\right) \leq b_n \exp\left(-c_n N \varepsilon^2\right) \quad .$$

Comments

- The acceptance/rejection comes (for usual SSM) from Douc et al. (2011);
- Olsson et al. (2017) have proposed the PaRIS algorithm : an *online* smoother (without backward pass) for additive functionals, i.e;

$$\mathbb{E}[H_n(X_{0:n}) | Y_{0:n}; \theta] = \sum_{k=1}^{n-1} \mathbb{E}[h_k(X_k, X_{k+1}) | Y_{0:n}; \theta]$$

The same trick provides an *online* version, resulting in the Generalized Random PaRIS alogrithm (GRand PaRIS);

GRand PaRIS is a hot topic in the French community

GRand PaRIS is a hot topic in the French community

Parisien POLITIQUE ÉCO SOCIÉTÉ FAITS DIVERS MA VILLE SPORTS LOISIRS

Tout ce que le Grand Paris va changer pour vous

Le GRAND Parisien | Frédéric Choulet et Jean-Gabriel Bontinck | 12 février 2018, 20h56 | MAJ : 13 février 2018, 0h17

ÇA ROULE

Président, financement... Le Grand Paris Express sur de nouveaux rails

Par Sibylle Vincendon — 7 septembre 2018 à 18:18

Le Parisien POLITIQUE ÉCO SOCIÉTÉ FAITS DIVERS MA VILLE SPORTS LOISIRS

Grand Paris : Valérie Pécresse regrette «une occasion ratée»

LA TRIBUNE ÉCONOMIE BOURSE ENTREPRISES & FINANCE HI-TECH VOS FINANCES IDÉES MÉTROPOLES CARRIÈRES

Opinions > Tribunes

Le Grand Paris vaut mieux qu'un jeu de mots

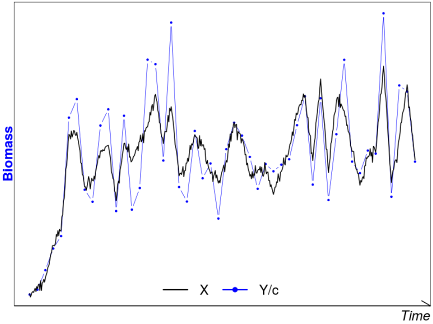
Par Par Alain Sarfati et Alain Cluzet | 28/09/2018, 16:03 | 1128 mots

Application for parameter estimation

Population dynamics model

$$\mathrm{d}X_t = \kappa X_t \left(1 - \frac{X_t}{\gamma}\right)\mathrm{d}t + \sigma X_t \mathrm{d}W_t, \quad X_0 = x_0 > 0, \qquad \text{Markov process}$$

$$Y_k = q X_k \exp(\varepsilon_k), \quad \varepsilon_k \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2_{obs}), \qquad \text{Observations}$$



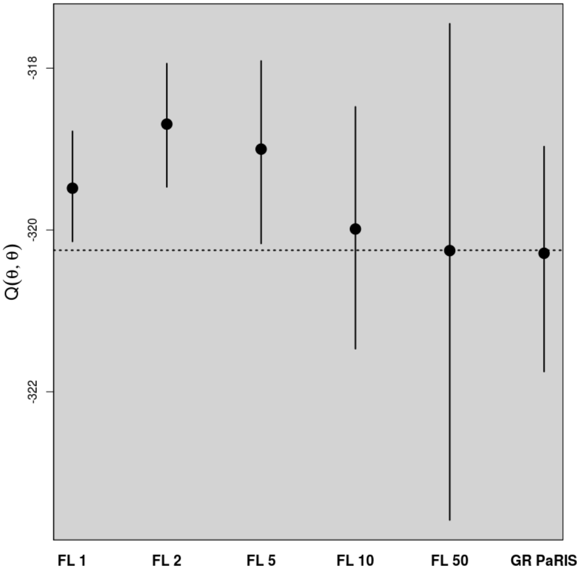
Objective: Approximation of the E step of an EM algorithm:

$$\mathcal{Q}(\theta_0, \theta) = \mathbb{E}[\overbrace{\ell(\theta; X_{0:n}, Y_{0:n})}^{H_n(X_{0:n})} \,|\, Y_{0:n}; \theta_0]$$

$$H_n(X_{0:n}) = \sum_{k=0}^{n-1} \log\left(q^\theta(X_k, X_{k+1}, \Delta_k) g^\theta_{k+1}(X_{k+1})\right)$$

Comparison with the fixed lag technique

Comparing our estimator with the fixed lag of Olsson and Strojby (2011)



Remark on the range of applications

Particle smoothers for POD:

- Olsson and Strojby (2011) and Gloaguen et al. (2018) both rely on Fearnhead et al. (2008) unbiased particle filtering (but this is a very active field of research);

Assumptions for the use of Fearnhead et al. (2008)

The hidden process satisfies $dX_t = \alpha^\theta(X_t)\mathrm{d}t + \beta^\theta(X_t)\mathrm{d}W_t$, if:

- **Lamperti transform** $\exists$ a 1-1 function η^θ s.t., for $\tilde{X}_s := \eta^\theta(X_s)$ satisfies

$$d\tilde{X}_s = \tilde{\alpha}^\theta(\tilde{X}_s)\mathrm{d}t + dW_t$$

- **Potential assumption** $\exists A^\theta : \mathbb{R} \mapsto \mathbb{R}$ s.t. $\tilde{\alpha}^\theta(x) = \nabla A^\theta(x)$;
- **Boundary assumption**

$$\lim_{\|x\|\to\infty} \| \tilde{\alpha}^\theta(x) \|^2 + \Delta A^\theta(x) < \infty \text{ and } \exists\ L \text{ s.t. } L \leq \| \tilde{\alpha}^\theta(x) \|^2 + \Delta A^\theta(x)$$

- Then a positive, a.s. bounded, and unbiased estimate of q^θ can be obtained;

Conclusions and perspectives

Conclusions

- New online smoother for SDE based SSMs;
- Mixes the tricks of Fearnhead et al. (2008), Douc et al. (2011) and Olsson et al. (2017);
- Asymptotically unbiased estimation, with at at best complexity of nN (at worse nN^2);
- Fit to the classical range of models for exact simulation algorithms of diffusion;
- Extending the range of SDE models? (Fearnhead et al., 2018).

References

Beskos, A., Papaspiliopoulos, O., Roberts, G., and Fearnhead, P. (2006). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes (with discusion). *J. Roy. Statist. Soc. Ser. B*, 68(3):333–382.

Douc, R., Garivier, A., Moulines, E., and Olsson, J. (2011). Sequential Monte Carlo smoothing for general state space hidden Markov models. *Ann. Appl. Probab.*, 21(6):2109–2145.

Fearnhead, P., Latuszynski, K., Roberts, G. O., and Sermaidis, G. (2018). Continious-time importance sampling: Monte carlo methods which avoid time-discretisation error. *arXiv preprint arXiv:1712.06201*.

Fearnhead, P., Papaspiliopoulos, O., and Roberts, G. (2008). Particle filters for partially observed diffusions. *J. Roy. Statist. Soc. Ser. B*, 70(4):755–777.

Gloaguen, P., Étienne, M.-P., and Le Corff, S. (2018). Online sequential monte carlo smoother for partially observed diffusion processes. *EURASIP Journal on Advances in Signal Processing*, 2018(1):9.

Gordon, N. J., Salmond, D. J., and Smith, A. F. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140-2, pages 107–113. IET.

Olsson, J. and Strojby, J. (2011). Particle-based likelihood inference in partially observed diffusion processes using generalised Poisson estimators. *Electron. J. Statist.*, 5:1090–1122.

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.

Proof of lemma

$$\mathbb{P}\left(Y = l | U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+}\right) = \frac{\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+} | Y = l\right)\mathbb{P}(Y = l)}{\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+}\right)}$$

$$= \frac{\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^l, \xi_{k+1}, \zeta)}{\sigma_+}\right)\omega_k^l}{\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+}\right)}$$

We now note that

$$\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+}\right) = \mathbb{E}_{p_\zeta}\left[\mathbb{E}\left[\mathbb{E}_{p_Y}\left[\mathbb{E}\left[\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+}\right) | Y\right]\right] | \zeta\right]\right]$$

$$= \mathbb{E}_{p_\zeta}\left[\sum_{l=1}^{N} w_k^l \frac{\hat{q}_k(\xi_k^l, \xi_{k+1}, \zeta)}{\sigma_+}\right] \quad \left(\text{As } \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+} \leq 1\right)$$

$$= \frac{1}{\sigma_+}\sum_{l=1}^{N} w_k^l q_k(\xi_k^l, \xi_{k+1}, \zeta)$$

In the same way, we have:

$$\mathbb{P}\left(U \leq \frac{\hat{q}_k(\xi_k^l, \xi_{k+1}, \zeta)}{\sigma_+}\right) = \frac{1}{\sigma_+} q_k(\xi_k^l, \xi_{k+1}, \zeta)$$

Which gives overall the wanted result

$$\mathbb{P}\left(Y = l | U \leq \frac{\hat{q}_k(\xi_k^Y, \xi_{k+1}, \zeta)}{\sigma_+}\right) = \frac{\omega_k^l q_k(\xi_k^l, \xi_{k+1}, \zeta)}{\sum_{l=1}^{N} w_k^l q_k(\xi_k^l, \xi_{k+1}, \zeta)} = p_X(l)$$