Sequential Monte Carlo samplers

Fixed dimensional state spaces

Introduction

- Many problems require sampling from

$$\pi_n(x) \propto \left(\prod_{k=0}^{n} G_k(x) \right) \pi_0(x)$$

  - Bayes posterior
    - product likelihood $G_k(x) = g(y_k|x)$
    - non-product likelihoods: $G_k = p(y|x)^{\phi_k - \phi_{k-1}}$ with $1 = \phi_n > \phi_{n-1} > \ldots > \phi_0 = 0$
  - rare event simulation:
    - estimate $\pi_0(A)$
    - $G_k = 1_{x \in A_k}$ with nested sets $A = A_n \subset \ldots \subset A_k \subset A_{k-1} \subset \ldots$
  - optimisation as sampling :
    - $\min_{x \in \mathcal{X}} V(x)$
    - $G_k = \exp(-\beta_k V)$ with $\beta_k > \beta_{k-1} > \ldots > 1$

Introduction

- In all these problems $x \in \mathcal{X}$, i.e. each π_n is defined on a fixed dimensional state space
- Sequential Monte Carlo (SMC)?
  - sampling using importance sampling and resampling
  - samples/particles propagated through sequence of distributions

$$\left(\Pi_n(x) = \frac{1}{Z_n}\gamma_n(x)\right)_{0 \leq n \leq T}$$

- <u>How can particle filters be used on a fixed dimensional state space ?</u>

Introduction

- Bayesian posterior for Hidden Markov models given observed data $y_{0:n}$:

$$\Pi_n(x_{0:n}) = p(x_{0:n} | y_{0:n}) = \frac{1}{Z_n} \prod_{k=0}^{n} f(x_k | x_{k-1}) G_k(x_k) \quad (1)$$

  and the *marginal likelihood*, $p(y_{0:n})$, is given by

$$Z_n = p(y_{0:n}) = \int \prod_{k=0}^{n} f(x_k | x_{k-1}) G_k(x_k) \, dx_{0:n}. \quad (2)$$

- Static problems

$$\pi_n(x) = \frac{1}{Z_n} \prod_{k=0}^{n} G_k(x) \pi_0(x)$$

  - can use same method using MCMC within SMC
  - use MCMC dynamics instead of $f(x|x')$

Outline

- Revision of particle filtering
- Generalising for static problems
- SMC samplers

Bayesian Filtering

- Often one wants the *marginal filter*

$$\pi_n(x_n) = \frac{1}{Z_n'} G_k(x_n) \int \pi_{n-1}(x_{n-1}) f(x_n | x_{n-1}) dx_{n-1}$$

- In practice algorithms operate on path space of $X_{0:n}$
- In most cases approximations of $\pi_n(x_n)$ are by-products or constructed based on approximations of $\Pi_n(x_{0:n})$

The bootstrap particle filter

For $n \geq 0$, (For $i = 1, \ldots N$)

1. Sample from dynamics

$$X_n^i \sim f(\cdot | \bar{X}_{n-1}^i)$$

2. Compute Importance weight:

$$\mathscr{W}_n^i \propto G_n(X_n^i), \quad \sum_{i=1}^{N} \mathscr{W}_n^i = 1.$$

3. Resample:

$$\bar{X}_{0:n}^i \sim \sum_{j=1}^{N} \mathscr{W}_{\mathrm{n}}^{\mathrm{i}} \delta_{X_{\mathbf{0}:n}^i}$$

- Particle approximations

$$\Pi_n^N = \sum_{j=1}^{N} \mathscr{W}_{\mathrm{n}}^{\mathrm{i}} \delta_{X_{\mathbf{0}:n}^i} \text{ or } \bar{\Pi}_n^N = \frac{1}{N} \sum_{j=1}^{N} \delta_{\bar{X}_{\mathbf{0}:n}^i}, \quad Z_n^N = Z_{n-1}^N \cdot \left(\frac{1}{N} \sum_{i=1}^{N} G_n(X_n^i) \right)$$

Sequential Importance Resampling

For $n \geq 0$, (For $i = 1, \ldots, N$)

1. Sample from dynamics

$$X_n^i \sim q(\cdot|\bar{X}_{n-1}^i, Y_n)$$

2. Compute Importance weight:

$$\mathscr{W}_n^i \propto \frac{G_n(X_n^i) f(X_n^i|\bar{X}_{n-1}^i)}{q(X_n^i|\bar{X}_{n-1}^i, Y_n)}, \quad \sum_{i=1}^{N} \mathscr{W}_n^i = 1.$$

3. Resample:

$$\bar{X}_{0:n}^i \sim \sum_{j=1}^{N} \mathscr{W}_{\mathbf{n}}^{\mathbf{i}} \delta_{X_{\mathbf{0}:n}^i}$$

Adaptive resampling

- Here we resample at every n
  - While resampling is a key component to have a good approximation it tends to leave early states in the path being represented by few particles.
- Better use resampling only when you need to
  - resample only when $ESS_n \leq \alpha N$
  - e.g. $\alpha = 1/2$.
  - here $ESS_n = \left(\sum_{i=1}^{N} \left(\mathscr{W}_n^i \right)^2 \right)^{-1}$
  - Note then expressions for $\mathscr{W}_n^i$ and Z_n^N change a bit.
- <u>Note estimate for Z_n^N is not unbiased in this case</u>

MCMC and particle filtering

- Similarly when filtering one can construct an MCMC kernel invariant to $\Pi_n(x_{0:n})$, say K_n, and then target

$$\tilde{\Pi}_n\left(dx_{0:n}, dx'_{0:n}\right) = \frac{1}{Z_n}\left(\prod_{k=0}^{n} f\left(x_k \mid x_{k-1}\right) G_k\left(x_k\right) dx_n\right) \cdot K_n(x_{0:n}, dx'_{0:n})$$

  - No problem as a both marginals (w.r.t to x or x') are the same!

$$\int \Pi_n\left(dx_{0:n}\right) K_n(x_{0:n}, dx'_{0:n}) = \Pi_n\left(dx'_{0:n}\right)$$

  - In addition, K_n can be designed just to move x_n not full $x_{0:n}$ (and this is still correct!)
    - this is the **resample move** of [Gillks and Berzuini 99]

SMC sampling

- Last idea is particulary useful when sampling from static target distributions
- Filtering for HMMs has a particular structure
  - dimension of target sequence increases,
  - likelihood is of product form

$$\Pi_n(x_{0:n}) \propto \left(\prod_{k=0}^{n} G_k\left(x_k\right) \right) Prior(x_{0:n})$$

  - Prior is Markov so one can sample sequentially $x_k | x_{k-1}$

SMC sampling

- Consider instead standard Bayes posterior

$$\pi_n(x) \propto \left(\prod_{k=0}^{n} G_k(x) \right) \pi_0(x)$$

- Recipe:
  - construct $\Pi_n(x_{0:n})$ admiting $\pi_n(x_n)$ as marginal
  - Set

    $$Prior(x_{0:n}) = \prod_{k=1}^{n} \mathcal{K}_k(x_{k-1}, x_k)$$

    using Markov dynamics with

    $$\pi_k \mathcal{K}_k = \pi_k$$

  - Chopin 01 Biometrika

Naive sampling

- Naive approach: sample $X_0^i \sim \pi_0$ and then using recursively
  1. Compute Importance weight:
  $$\mathscr{W}_n^i \propto G_n(X_{0,n}^i), \quad \sum_{i=1}^{N} \mathscr{W}_n^i = 1.$$
  2. Resample:
  $$\bar{X}_{0,n}^i \sim \sum_{j=1}^{N} \mathscr{W}_{\mathrm{n}}^{\mathrm{i}} \delta_{X_{\mathbf{0}:n}^i}$$
- Path degeneracy means this method will degenerate
  - exploration happens only at X_0, no diversity reintroduced
  - Monte Carlo variance will increase very fast

SMC sampling

- More sensible approach (Chopin 01)
  1. Compute Importance weight:

$$\mathscr{W}_n^i \propto G_n(X_n^i), \quad \sum_{i=1}^{N} \mathscr{W}_n^i = 1.$$

  2. Resample and move:

$$\bar{X}_n^i \sim \sum_{j=1}^{N} \mathscr{W}_\mathrm{n}^\mathrm{i} \mathcal{K}_n \left(X_n^i, \cdot\right)$$

- $\mathcal{K}_n$ here is a MCMC kernel: $\pi_n \mathcal{K}_n = \pi_n$
  - diversity is reintroduced in a way that statistical properties of sample are maintained

SMC sampling

There are many advantages here compared to standard MCMC.

- One can tune proposals using statistics from π_n^N
  - this can lead to good mixing
- One can monitor output and performance as n increases
  - this can lead to adaptive algorithms (tempering for each G_n, adapting number of MCMC iterations,...)
- Robust to Multimodality
- Normalising const. estimation is easier
  - unbiased (for non adaptive SMC)
- Many steps are easy to parallelise

Discussion

- $\mathcal{K}_n$ being π_n invariant is not only possible choice
  - can always correct using IS
  - Section 2.3 in [Del Moral, Doucet, Jasra 2006]
- Convergence follows similarly to before
  - consistent, CLT, ...
  - if likelihood follows some asymptotic normality in n then finite number of resampling steps are needed (when adaptive resampling) [Chopin 01, preprint]
- Understanding from particle filters still applies
  - if π_n is very different from π_{n-1} variance of weights can be large even if $\mathcal{K}_n$ mixes fast.

Path space reformulation

- Can couple π_n with π_{n-1} and previous marginals in many possible ways
- Consider the following target distribution for time n

$$\Pi_n(dx_{0:n}) = \pi_n(dx_n) \prod_{k=n-1}^{1} L_k(x_{k+1}, dx_k)$$

  - L_k is an arbitrary backward Markov kernel
- Recall

$$\Pi_n(dx_{0:n}) = \frac{1}{Z_n}\gamma_n(x_{0:n})dx_{0:n}$$

  and let for now

$$\gamma_n(x_{0:n}) = \gamma_n(x_n) \prod_{k=n-1}^{1} L_k(x_{k+1}, x_k)$$

Path space reformulation

- As in filtering will use IS with Markov proposal with density

$$\eta_n(x_{0:n}) = \pi_0(x_0) \prod_{k=1}^{n-1} K_k(x_{k-1}, x_k)$$

  - K_k is an arbitrary forward Markov proposal
  - ... but some choices are better than others

Path space reformulation

- Recursive weight becomes

$$w_n(x_{0:n}) = w_{n-1}(x_{0:n-1}) \underbrace{\frac{\gamma_n(x_n) L_{n-1}(x_n, x_{n-1})}{\gamma_{n-1}(x_{n-1}) K_n(x_{n-1}, x_n)}}_{\tilde{w}(x_{n-1}, x_n)}$$

- $\tilde{w}(x_{n-1}, x_n)$ is incremental weight used in algorithm
  - shaped both by L_k and K_k
  - Del Moral et. al. 06 explores design of L_k in detail
    - optimal, suboptimal, convenient when K_k is MCMC kernel, mixtures

SMC sampler

For $n \geq 0$, (For $i = 1, \ldots N$)

1. Sample from dynamics

$$X_n^i \sim K_n(\bar{X}_{n-1}^i, \cdot)$$

2. Compute Importance weight:

$$\mathscr{W}_n^i \propto \tilde{w}_n(\bar{X}_{n-1}^i, X_n^i), \quad \sum_{i=1}^{N} \mathscr{W}_n^i = 1.$$

3. Resample:

$$\bar{X}_{0:n}^i \sim \sum_{j=1}^{N} \mathscr{W}_{\mathbf{n}}^{\mathbf{i}} \delta_{X_{\mathbf{0}:n}^i}$$

SMC approximations

- Particle approximations

$$\Pi_n^N = \sum_{j=1}^{N} \mathscr{W}_{\mathrm{n}}^{\mathrm{i}} \delta_{X_{\mathbf{0}:n}^i} \text{ or } \bar{\Pi}_n^N = \frac{1}{N} \sum_{j=1}^{N} \delta_{\bar{X}_{\mathbf{0}:n}^i}$$

and

$$Z_n^N = Z_{n-1}^N \cdot \left(\frac{1}{N} \sum_{i=1}^{N} \tilde{w}_n(\bar{X}_{n-1}^i, X_n^i) \right)$$

A useful extension: adaptive tempering

- if π_n is very different from π_{n-1} can also temper and bridge gap:

$$\pi_{n,m} \propto G_n^{\phi_{n,m}} \pi_{n-1}$$

- Can tune adaptively by solving bisection:

$$\phi_{n,m+1} = \inf \left\{ \phi : \ \phi_m < \phi \leq 1, : ESS^N \left(\Pi^N_{n,m+1}, \phi \right) = N_{thresh} \right\}$$

  - here $ESS^N = (\sum_{j=1}^N \left(\mathscr{W}^j_{n,m+1} \right)^2)^{-1}$

- See
  - [Beskos, Jasra, K., Thiery 16] or [Cerou & Guyader 16] for some convergence results.
  - [Zhou, Johansen, Aston 16] for extensions.

SMC^2: parameter inference for HMMs

- Recall in HMMs

$$p\left(\theta \mid y_{0:n}\right) \propto \prod_{k=0}^{n} p_\theta\left(y_k \mid y_{0:k-1}\right) p(\theta).$$

- For fixed θ one can use a PF approximations:

$$\widehat{p}_\theta\left(y_k \mid y_{0:k-1}\right) = \frac{1}{N_x} \sum_{i=1}^{N_x} w_k^\theta(X_{k-1,k}^i)$$

- SMC^2 is a SMC sampler for θ with particles $\left\{\theta^l\right\}_{l=1}^{N_\theta}$
  - for each θ_n^l use $\widehat{p}_{\theta_n^l}\left(y_k \mid y_{0:k-1}\right)$ obtained from a PF with N_x particles for $x_{0:n}$
  - Move θ_n^l with PMCMC kernels invariant to

$$p\left(\left\{\left\{x_n^{i,l}, o_n(i,l)\right\}_{i=1}^{N_x}\right\}_{n=1}^{T}, \theta_n^l \,\middle|\, y_{0:n}\right)$$

- See [Chopin, Jacob, Papaspiliopoulos 13].

Discussion

- Here one relies on absolute continuity of $\pi_n(dx_n)L_{n-1}(x_n, dx_{n-1})$ with $\pi_{n-1}(dx_{n-1})K_n(x_{n-1}, dx_n)$
  - one could use contructions for direct approach of $\pi_{n-1}K_n$ targetting π_n when K_n is MCMC kernel
  - see note in A. Doucet's website
- Approach has been very powerful in applications
  - with different structure for Π_n or state space
  - statistics: inference, ABC, phylogenetics, ...
  - beyond: high dimensional inverse problems, computational physics, rare event simulation, optimisation
  - adapting many steps even target distributions very useful

Reading list

- Portal by Arnaud Doucet:
  `http://www.stats.ox.ac.uk/~doucet/smc_resources.html`

Papers

- Chopin (2001) A Sequential Particle Filter Method for Static Models, Biometrika
- Del Moral, Doucet, Jasra (2006) Sequential Monte Carlo Samplers, JRSSB.
- Zhou, Johansen, Aston (2016) Toward automatic model comparison: an adaptive sequential Monte Carlo approach JCGS
- Chopin, Jacob, & Papaspiliopoulos, O. (2013). SMC2: an efficient algorithm for sequential analysis of state space models. JRSSB