SMC^2: an efficient algorithm for sequential analysis of state-space models

N. CHOPIN[1], P.E. JACOB[2], & O. PAPASPILIOPOULOS[3]

[1] ENSAE-CREST

[2] CREST & Université Paris Dauphine,

[3] Universitat Pompeu Fabra & Ramón y Cajal

Statement of the problem

State Space Models

A system of equations

- Hidden states (Markov): $p(x_1|\theta) = \mu_\theta(x_1)$ and for $t \geq 1$

$$p(x_{t+1}|x_{1:t}, \theta) = p(x_{t+1}|x_t, \theta) = f_\theta(x_{t+1}|x_t)$$

- Observations:

$$p(y_t|y_{1:t-1}, x_{1:t-1}, \theta) = p(y_t|x_t, \theta) = g_\theta(y_t|x_t)$$

- Parameter: $\theta \in \Theta$, prior $p(\theta)$.

We observe $y_{1:T} = (y_1, \ldots y_T)$, T might be large ($\approx 10^4$). x and θ also of several dimensions. Many models where f_θ or g_θ cannot be written in closed form

State Space Models

Some interesting distributions

Bayesian inference focuses on:

$$\text{static: } p(\theta|y_{1:T}) \quad \text{dynamic: } p(\theta|y_{1:t})\,, t \in 1:T$$

Filtering (traditionally) focuses on:

$$\forall t \in [1, T] \quad p_\theta(x_t|y_{1:t})$$

Smoothing (traditionally) focuses on:

$$\forall t \in [1, T] \quad p_\theta(x_t|y_{1:T})$$

Prediction:

$$\begin{aligned} p(y_{t+1} \mid y_{1:t}) &= \int g_\theta(y_{t+1} \mid x_{t+1}) f_\theta(x_{t+1} \mid x_t) \\ &\times \quad p_\theta(x_t \mid y_{1:t}) p(\theta \mid y_{1:t}) dx_t d\theta \end{aligned}$$

An example

Stochastic Volatility (Lévy-driven models)

- Observations ("log returns"):

$$y_t = \mu + \beta v_t + v_t^{1/2} \epsilon_t \, , t \geq 1$$

- Hidden states ("actual volatility" - integrated process):

$$v_{t+1} = \frac{1}{\lambda}(z_t - z_{t+1} + \sum_{j=1}^{k} e_j)$$

An example

…where the process z_t is the "spot volatility":

$$z_{t+1} = e^{-\lambda} z_t + \sum_{j=1}^{k} e^{-\lambda(t+1-c_j)} e_j$$

$$k \sim \mathrm{Poi}\left(\lambda\xi^2/\omega^2\right) \quad c_{1:k} \overset{iid}{\sim} \mathrm{U}(t, t+1) \quad e_{i:k} \overset{iid}{\sim} \mathrm{Exp}\left(\xi/\omega^2\right)$$

The parameter is $\theta \in (\mu, \beta, \xi, \omega^2, \lambda)$, and $x_t = (v_t, z_t)'$.

▸ See the results

Why are those models challenging?

. . . It is effectively impossible to compute the likelihood

$$p(y_{1:T}|\theta) = \left[\int_{\mathcal{X}^T} p(y_{1:T}|x_{1:T}, \theta) p(x_{1:T}|\theta) dx_{1:T} \right]$$

Similarly, all other inferential quantities are impossible to compute.

Problems with MCMC and IS approaches

- We cannot compute the likelihood or sample from $p(\theta|y_{1:T})$ directly
- Importance Sampling (IS) results in polynomial or exponential in t increase of variance; e.g. Section 4 in Kong, Liu, Wong (1994, JASA); Bengtsson, T., Bickel, P., and Li, B. (2008); Theorem 4 of Chopin (2004; AoS)

- An MCMC approach would be to sample the parameters and states jointly. But:
- The high-dimensional autocorrelated state process is difficult to simulate efficiently conditionally on the observations
- High dependence between parameters and state variables cause poor performance of Gibbs sampler
- Furthermore, MCMC methods are not designed to recover the whole sequence $\pi(x_{1:t}, \theta \mid y_{1:t})$ computationally efficiently, instead they sample from the "static" distribution $\pi(x_{1:T}, \theta \mid y_{1:T})$

Methodology Pt I

Monte Carlo in Bayesian statistics: the new generation

This problem is an instance of situation frequently encountered in modern applications: statistical inference with intractable densities, which however can be estimated using Monte Carlo.

This is harder version of the main problem in Bayesian statistics (intractable integrals) which traditionally has been addressed by data augmentation, simulated likelihood, EM, etc.

A new generation of methods has emerged, a common feature of which is the interplay between unbiased estimation and auxiliary variables with aim at constructing exact MC

Some characteristic references in that respect include:

- ABC methods, e.g Pritchard et. al (1999), Molec. Biol. Evol
- MCMC for models with intractable normalizing constants, e.g Møller et. al (2006), Biometrika
- Pseudo-marginal MCMC, Andrieu and Roberts (2009), Ann. Stat.
- Random weight particle filters, e.g Fearnhead et. al (2010), J. Roy. Stat. Soc. B
- Particle MCMC methods, e.g Andrieu, Doucet and Holenstein (2010), J. Roy. Stat. Soc. B

Our approach fits in this framework, allowing fully Bayesian sequential inference (whereas aforementioned approaches deal with "static" MCMC or inference for dynamic models with fixed parameters)

Main tools of our approach

- Particle filter algorithms for state-space models
- Iterated Batch Importance Sampling for sequential Bayesian inference for parameters

Both are sequential Monte Carlo (SMC) methods

1. Particle filters

Consider the simplified problem of targeting

$$p_\theta(x_{t+1}|y_{1:t+1})$$

for a given value of θ.

This sequence of distributions is approximated by a sequence of weighted particles which are properly weighted using importance sampling, mutated/propagated according to the system dynamics, and resampled to control the variance.

Below we give a pseudo-code version. Any operation involving the superscript n must be understood as performed for $n = 1 : N_x$, where N_x is the total number of particles.

Step 1: At iteration $t = 1$,

(a) Sample $x_1^n \sim q_{1,\theta}(\cdot)$.

(b) Compute and normalise weights

$$w_{1,\theta}(x_1^n) = \frac{\mu_\theta(x_1^n) g_\theta(y_1|x_1^n)}{q_{1,\theta}(x_1^n)}, \quad W_{1,\theta}^n = \frac{w_{1,\theta}(x_1^n)}{\sum_{i=1}^N w_{1,\theta}(x_1^i)}.$$

Step 2: At iteration $t = 2 : T$

(a) Sample the index $a_{t-1}^n \sim \mathcal{M}(W_{t-1,\theta}^{1:N_x})$ of the ancestor

(b) Sample $x_t^n \sim q_{t,\theta}(\cdot|x_{t-1}^{a_{t-1}^n})$.

(c) Compute and normalise weights

$$w_{t,\theta}(x_{t-1}^{a_{t-1}^n}, x_t^n) = \frac{f_\theta(x_t^n|x_{t-1}^{a_{t-1}^n}) g_\theta(y_t|x_t^n)}{q_{t,\theta}(x_t^n|x_{t-1}^{a_{t-1}^n})}, \quad W_{t,\theta}^n = \frac{w_{t,\theta}(x_{t-1}^{a_{t-1}^n}, x_t^n)}{\sum_{i=1}^{N_x} w_{t,\theta}(x_{t-1}^{a_{t-1}^i}, x_t^i)}$$

Observations

- At each t, $(w_t^{(i)}, x_{1:t}^{(i)})_{i=1}^{N_x}$ is a particle approximation of $p_\theta(x_t|y_{1:t})$.
- Resampling to avoid degeneracy.
- In principle $(w_t^{(i)}, x_{1:t}^{(i)})_{i=1}^{N_x}$ is also a particle approximation of $p_\theta(x_{1:t}|y_{1:t})$ (bad notation! careful with genealogy)
- Resampling makes this a very poor approximation for large t, known as the **path degeneracy problem**
- Taking $q_\theta = f_\theta$ simplifies weights, but mainly yields a feasible algorithm when f_θ can only be simulated
- **Notation:** $\psi_{t,\theta}$ the distribution that all variables are drawn from upto time t (particles and ancestors)

Unbiased likelihood estimator

A by-product of PF output is that

$$\hat{Z}_t^N = \prod_{t=1}^{T} \left(\frac{1}{N_x} \sum_{i=1}^{N_x} w_t^{(i)} \right)$$

is an *unbiased* estimator of the likelihood $Z_t = p(y_{1:t}|\theta)$ for all t.

Whereas consistency of the estimator is immediate to check, unbiasedness is **subtle**, see e.g Proposition 7.4.1 in Del Moral. The variance of this estimator grows typically linealy with T (and not exponentially) because of **dependence** of the factors in the product.

2. IBIS

SMC method for particle approximation of the sequence $p(\theta \mid y_{1:t})$ for $t = 1 : T$, for models where likelihood is tractable; see e.g. Chopin (2002, Bmka) for details

Again, the sequence of parameter posterior distribution is approximated by N_θ weighted particles,

$$(\theta^m, \omega^m)_{m=1}^{N_\theta}$$

By product: consistent estimators of the predictive densities

$$L_t = \int p(y_t | y_{1:t-1}, \theta) p(\theta) d\theta$$

hence of the model evidence

In the next slide we give the pseudo-code of the IBIS algorithm. Operations with superscript m must be understood as operations performed for all $m \in 1 : N_\theta$

Sample θ^m from $p(\theta)$ and set $\omega^m \leftarrow 1$. Then, at time $t = 1, \ldots, T$

(a) Compute the incremental weights and their weighted average

$$u_t(\theta^m) = p(y_t|y_{1:t-1}, \theta^m), \quad \hat{L}_t = \frac{1}{\sum_{m=1}^{N_\theta} \omega^m} \times \sum_{m=1}^{N_\theta} \omega^m u_t(\theta^m),$$

(b) Update the importance weights,

$$\omega^m \leftarrow \omega^m u_t(\theta^m). \tag{1}$$

(c) If some degeneracy criterion is fulfilled, sample $\tilde{\theta}^m$ independently from the mixture distribution

$$\frac{1}{\sum_{m=1}^{N_\theta} \omega^m} \sum_{m=1}^{N_\theta} \omega^m K_t\left(\theta^m, \cdot\right).$$

Finally, replace the current weighted particle system:

$$(\theta^m, \omega^m) \leftarrow (\tilde{\theta}^m, 1).$$

Observations

- Cost of lack of ergodicity in θ: the occasional MCMC move
- Still, in regular problems resampling happens at diminishing frequency (logarithmically)
- K_t is an MCMC kernel invariant wrt $\pi(\theta \mid y_{1:t})$. Its parameters can be chosen using information from current population of θ-particles
- $\hat{L}_t$ can be used to obtain an estimator of the model evidence
- Infeasible to implement for state-space models: intractable incremental weights, and MCMC kernel

Our algorithm: SMC^2

We provide a generic (black box) algorithm for recovering the sequence of parameter posterior distributions, but as well filtering, smoothing and predictive.

We give next a pseudo-code; the code seems to only track the parameter posteriors, but actually it does all other jobs. Superficially, it looks an approximation of IBIS, but in fact it **does not produce any systematic errors** (unbiased MC)

Sample θ^m from $p(\theta)$ and set $\omega^m \leftarrow 1$. Then, at time $t = 1, \ldots, T$,

(a) For each particle θ^m, perform iteration t of the PF: If $t = 1$, sample independently $x_1^{1:N_x,m}$ from ψ_{1,θ^m}, and compute

$$\hat{p}(y_1|\theta^m) = \frac{1}{N_x}\sum_{n=1}^{N_x} w_{1,\theta}(x_1^{n,m});$$

If $t > 1$, sample $\left(x_t^{1:N_x,m}, a_{t-1}^{1:N_x,m}\right)$ from ψ_{t,θ^m} conditional on $\left(x_{1:t-1}^{1:N_x,m}, a_{1:t-2}^{1:N_x,m}\right)$, and compute

$$\hat{p}(y_t|y_{1:t-1},\theta^m) = \frac{1}{N_x}\sum_{n=1}^{N_x} w_{t,\theta}(x_{t-1}^{a_{t-1}^{n,m},m}, x_t^{n,m}).$$

(b) Update the importance weights,

$$\omega^m \leftarrow \omega^m \hat{p}(y_t | y_{1:t-1}, \theta^m)$$

(c) If some degeneracy criterion is fulfilled, sample $\left(\tilde{\theta}^m, \tilde{x}_{1:t}^{1:N_x,m}, \tilde{a}_{1:t-1}^{1:N_x}\right)$ independently from

$$\frac{1}{\sum_{m=1}^{N_\theta} \omega^m} \sum_{m=1}^{N_\theta} \omega^m K_t \left\{ \left(\theta^m, x_{1:t}^{1:N_x,m}, a_{1:t-1}^{1:N_x,m} \right), \cdot \right\}$$

Finally, replace current weighted particle system:

$$(\theta^m, x_{1:t}^{1:N_x,m}, a_{1:t-1}^{1:N_x,m}, \omega^m) \leftarrow (\tilde{\theta}^m, \tilde{x}_{1:t}^{1:N_x,m}, \tilde{a}_{1:t-1}^{1:N_x,m}, 1)$$

Observations

- It appears as approximation to IBIS. For $N_x = \infty$ it is IBIS.
- However, no approximation is done whatsoever. This algorithm really samples from $p(\theta|y_{1:t})$ and all other distributions of interest. One would expect an increase of MC variance over IBIS.
- The validity of algorithm is essentially based on two results: i) the particles are **properly weighted** due to unbiasedness of PF estimator of likelihood; ii) the MCMC kernel is appropriately constructed to maintain invariance wrt to an **expanded distribution** which admits those of interest as marginals; it is a Particle MCMC kernel.
- The algorithm does not suffer from the path degeneracy problem due to the MCMC updates

Theory Pt I

Proposition

The probability density π_t may be written as:

$$\begin{aligned}
\pi_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x}) &= p(\theta|y_{1:t}) \\
&\times \quad \frac{1}{N_x}\sum_{n=1}^{N_x} \frac{p(\mathbf{x}_{1:t}^n|\theta, y_{1:t})}{N_x^{t-1}} \left\{ \prod_{\substack{i=1 \\ i\neq \mathbf{h}_t^n(1)}}^{N_x} q_{1,\theta}(x_1^i) \right\} \\
&\times \quad \left\{ \prod_{s=2}^{t} \prod_{\substack{i=1 \\ i\neq \mathbf{h}_t^n(s)}}^{N_x} W_{s-1,\theta}^{a_{s-1}^i} q_{s,\theta}(x_s^i|x_{s-1}^{a_{s-1}^i}) \right\}
\end{aligned}$$

▸ Intuition when $t = 1$

Methodology Pt II

Principled framework for algorithmic improvements

Elaborating on Proposition 1 we propose several formal ways for the following algorithmic operations:

- MCMC rejuvenation step within the PMCMC
- Sampling from the smoothing distributions
- Automatic calibration of N_x
- Dealing with intractable g_θ
- SMC^2 can be used for more general sequences of distributions, e.g. obtained by tempering

The MCMC rejuvenation step

(a) Sample $\tilde{\theta}$ from proposal kernel, $\tilde{\theta} \sim T(\theta, d\tilde{\theta})$.

(b) Run a new PF for $\tilde{\theta}$: sample independently $(\tilde{x}_{1:t}^{1:N_x}, \tilde{a}_{1:t-1}^{1:N_x})$ from $\psi_{t,\tilde{\theta}}$, and compute $\hat{Z}_t(\tilde{\theta}, \tilde{x}_{1:t}^{1:N_x}, \tilde{a}_{1:t-1}^{1:N_x})$.

(c) Accept the move with probability

$$1 \wedge \frac{p(\tilde{\theta})\hat{Z}_t(\tilde{\theta}, \tilde{x}_{1:t}^{1:N_x}, \tilde{a}_{1:t-1}^{1:N_x})\, T(\tilde{\theta}, \theta)}{p(\theta)\hat{Z}_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x})\, T(\theta, \tilde{\theta})}\,.$$

It directly follows from the Proposition that this algorithm defines a standard Hastings-Metropolis kernel with proposal distribution

$$q_\theta(\tilde{\theta}, \tilde{x}_{1:t}^{1:N_x}, \tilde{a}_{1:t}^{1:N_x}) = T(\theta, \tilde{\theta}) \psi_{t,\tilde{\theta}}(\tilde{x}_{1:t}^{1:N_x}, \tilde{a}_{1:t}^{1:N_x})$$

and admits as invariant distribution the extended distribution $\pi_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x})$.

This is precisely a particle MCMC step, as in Andrieu et al. (2010, JRSSB).

Dynamic increase of N_x

Why increasing N_x is necessary?

Our framework allows the dynamic and automatic increase of N_x using the generalized importance sampling strategy of Del Moral et al. (2006).

We propose two approaches; a particle exchange and a conditional SMC, the latter being more efficient (in terms of minimizing variance of weights) but memory intensive

They get triggered when a degeneracy criterion is fulfilled

Particle exchange

Exchange importance sampling step

Launch a new SMC for each θ-particle, with $\tilde{N}_x$ x-particles. Joint distribution:

$$\pi_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x})\psi_{t,\theta}(\tilde{x}_{1:t}^{1:\tilde{N}_x}, \tilde{a}_{1:t-1}^{1:\tilde{N}_x})$$

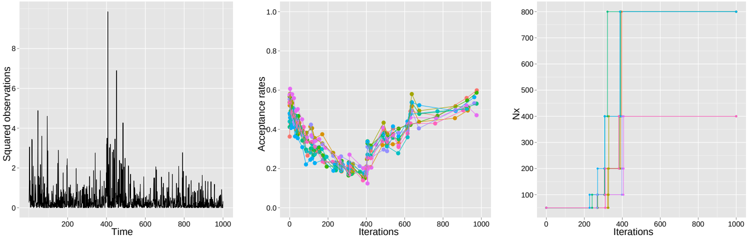
Retain the new x-particles and drop the old ones, updating the θ-weights with:

$$u_t^{exch}\left(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x}, \tilde{x}_{1:t}^{1:\tilde{N}_x}, \tilde{a}_{1:t-1}^{1:\tilde{N}_x}\right) = \frac{\hat{Z}_t(\theta, \tilde{x}_{1:t}^{1:\tilde{N}_x}, \tilde{a}_{1:t-1}^{1:\tilde{N}_x})}{\hat{Z}_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x})}$$

Numerics

Extensive study of numerical performance and comparisons, both in the paper and its Supplement, both available at ArXiv.

Numerical illustrations: SV

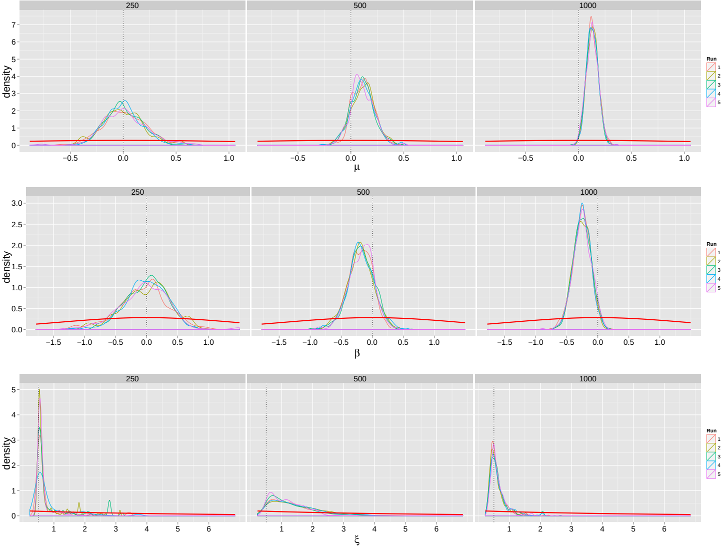


(a) (b) (c)

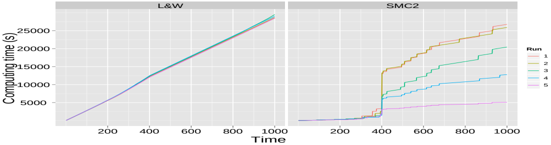
Figure: Squared observations (synthetic data set), acceptance rates, and illustration of the automatic increase of N_x.

▸ See the model

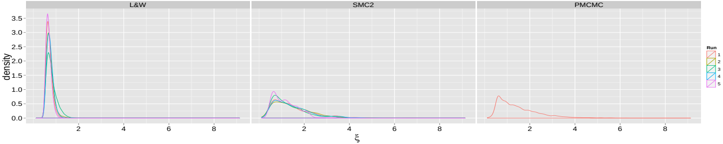
Numerical illustrations: SV



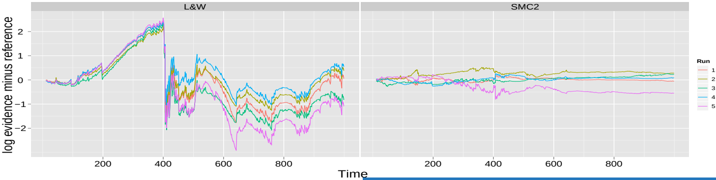
Numerical illustrations: SV



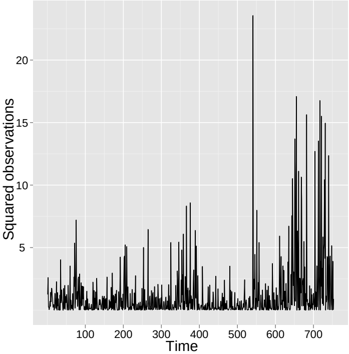
(a)



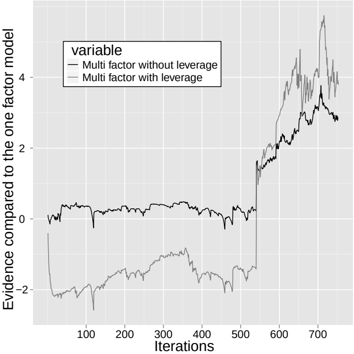
(b)



Numerical illustrations: SV



(d)



(e)

Theory Pt II

Stability of the algorithm and computational complexity

The algorithm refreshes at each MCMC step, but at the expense of re-visiting all available data so far. The computational complexity of the algorithm is determined by the frequency at which needs to resort to this step.

If after each resampling the particles are simulated from π_t, then for a time $t+p$ such that no resampling has happened since t, the inverse of the second moment of the normalized weights in SMC^2 and IBIS is given by

$$\mathcal{E}^{N_x}_{t,t+p} = \left\{ \mathbb{E}_{\bar{\pi}_{t,t+p}} \left[\frac{\hat{Z}_{t+p|t}(\theta, x^{1:N_x}_{1:t+p}, a^{1:N_x}_{1:t+p-1})^2}{p(y_{t+1:t+p}|y_{1:t})^2} \right] \right\}^{-1}$$

$$\mathcal{E}^{\infty}_{t,t+p} = \left\{ \mathbb{E}_{p(\theta|y_{1:t})} \left[\frac{p(\theta|y_{1:t+p})^2}{p(\theta|y_{1:t})^2} \right] \right\}^{-1}$$

Proposition

1. *Under Assumptions (H1a) and (H1b) in* ▸ Appendix *there exists a constant* $\eta > 0$ *such that for any* p*, if* $N_x > \eta p$*,*

$$\mathcal{E}_{t,t+p}^{N_x} \geq \frac{1}{2}\mathcal{E}_{t,t+p}^{\infty}. \tag{2}$$

2. *Under Assumptions (H2a)-(H2d) in* ▸ Appendix *for any* $\gamma > 0$ *there exist* $\tau, \eta > 0$ *and* $t_0 < \infty$*, such that for* $t \geq t_0$*,*

$$\mathcal{E}_{t,t+p}^{N_x} \geq \gamma, \text{ for } p = \lceil \tau t \rceil, N_x = \lceil \eta t \rceil.$$

This suggests that the computational cost of the algorithm up to time t is $\mathcal{O}(N_\theta t^2)$ which should be contrasted with the $\mathcal{O}(N_\theta t)$ cost of IBIS under the Assumptions (H2a)-(H2d).

Final Remarks

A powerful framework

- The article is available on arXiv and our web pages
- A package is available:

  http://code.google.com/p/py-smc2/.

- Parallel computing implementation using GPUs by Fulop and Li (2011)

Appendix

Some challenges for short-term future

- Theoretical: justify the ▸ orange claim
- Numerical: combined with GPU implementation, try the algorithm on extremely hard problems
- Algorithmic: find a better diagnostic for increase N_x

Why does it work? - Intuition for $t = 1$

At time $t = 1$, the algorithm generates variables θ^m from the prior $p(\theta)$, and for each θ^m, the algorithm generates vectors $x_1^{1:N_x,m}$ of particles, from $\psi_{1,\theta^m}(x_1^{1:N_x})$.

Thus, the sampling space is $\Theta \times \mathcal{X}^{N_x}$, and the actual "particles" of the algorithm are N_θ independent and identically distributed copies of the random variable $(\theta, x_1^{1:N_x})$, with density:

$$p(\theta)\psi_{1,\theta}(x_1^{1:N_x}) = p(\theta) \prod_{n=1}^{N_x} q_{1,\theta}(x_1^n).$$

Then, these particles are assigned importance weights corresponding to the incremental weight function $\hat{Z}_1(\theta, x_1^{1:N_x}) = N_x^{-1} \sum_{n=1}^{N_x} w_{1,\theta}(x_1^n)$.

This means that, at iteration 1, the target distribution of the algorithm should be defined as:

$$\pi_1(\theta, x_1^{1:N_x}) = p(\theta)\psi_{1,\theta}(x_1^{1:N_x}) \times \frac{\hat{Z}_1(\theta, x_1^{1:N_x})}{p(y_1)},$$

where the normalising constant $p(y_1)$ is easily deduced from the property that $\hat{Z}_1(\theta, x_1^{1:N_x})$ is an unbiased estimator of $p(y_1|\theta)$.

Direct substitutions yield

$$\begin{aligned}
\pi_1(\theta, x_1^{1:N_x}) &= \frac{p(\theta)}{p(y_1)} \prod_{i=1}^{N_x} q_{1,\theta}(x_1^i) \left\{ \frac{1}{N_x} \sum_{n=1}^{N_x} \frac{\mu_\theta(x_1^n) g_\theta(y_1|x_1^n)}{q_{1,\theta}(x_1^n)} \right\} \\
&= \frac{1}{N_x} \sum_{n=1}^{N_x} \frac{p(\theta)}{p(y_1)} \mu_\theta(x_1^n) g_\theta(y_1|x_1^n) \left\{ \prod_{i=1, i \neq n}^{N_x} q_{1,\theta}(x_1^i) \right\}
\end{aligned}$$

and noting that, for the triplet (θ, x_1, y_1) of random variables,

$$p(\theta)\mu_\theta(x_1) g_\theta(y_1|x_1) = p(\theta, x_1, y_1) = p(y_1) p(\theta|y_1) p(x_1|y_1, \theta)$$

one finally gets that:

$$\pi_1(\theta, x_1^{1:N_x}) = \frac{p(\theta|y_1)}{N_x} \sum_{n=1}^{N_x} p(x_1^n|y_1, \theta) \left\{ \prod_{i=1, i \neq n}^{N_x} q_{1,\theta}(x_1^i) \right\}.$$

By a simple induction, one sees that the target density π_t at iteration $t \geq 2$ should be defined as:

$$\pi_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x}) = p(\theta)\psi_{t,\theta}(x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x}) \times \frac{\hat{Z}_t(\theta, x_{1:t}^{1:N_x}, a_{1:t-1}^{1:N_x})}{p(y_{1:t})}$$

and similarly we obtain Proposition 1

Assumptions

(H1a) For all $\theta \in \Theta$, and $x, x', x'' \in \mathcal{X}$,

$$\frac{f_\theta(x|x')}{f_\theta(x|x'')} \leq \beta.$$

(H1b) For all $\theta \in \Theta$, $x, x' \in \mathcal{X}$, $y \in \mathcal{Y}$,

$$\frac{g_\theta(y|x)}{g_\theta(y|x')} \leq \delta.$$

- (H2a) The MLE $\hat{\theta}_t$ (the mode of function $l_t(\theta)$) exists and converges to θ_0 as $n \to +\infty$.
- (H2b) The observed information matrix defined as

$$\Sigma_t = -\frac{1}{t}\frac{\partial l_t(\hat{\theta}_t)}{\partial\theta\partial\theta'}$$

  is positive definite and converges to $I(\theta_0)$, the Fisher information matrix.
- (H2c) There exists Δ such that, for $\delta \in (0, \Delta)$,

$$\limsup_{t\to+\infty}\left[\frac{1}{t}\sup_{\|\theta-\hat{\theta}_t\|>\delta}\left\{l_t(\theta) - l_t(\hat{\theta}_t)\right\}\right] < 0.$$

- (H2d) The function l_t/t is six-times continuously differentiable, and its derivatives of order six are bounded relative to t over any compact set $\Theta' \subset \Theta$.