Optional exercises for particle marginal methods

MVE187 / MSA101 Computational methods for Bayesian statistics

Notice the following exercises are optional, for the interested students. I am aware that they are not trivial, as the material introduced at lecture requires to be digested and implementing things takes time. I am happy to get questions, I can setup individual meetings in person or via zoom, just let me know at picchini@chalmers.se

At lecture we have considered the following state space model:

$$\begin{cases} x_t = 0.5x_{t-1} + 25\frac{x_{t-1}}{(1+x_{t-1}^2)} + 8\cos(1.2(t-1)) + v_t, \\ y_t = 0.05x_t^2 + e_t, \end{cases}$$

with deterministic $x_0 = 0$ and $v_t \sim \mathcal{N}(0, q)$ (i.i.d), $e_t \sim \mathcal{N}(0, r)$ (i.i.d) and v_t independent of e_t for any t. Notice **here q and r are variances not standard deviations**. At lecture we considered 100 data points generated at times $t = \{1, 2, ..., 100\}$ using $\theta = (q, r) = (0.1, 1)$, and performed Bayesian inference on θ.

Not all questions refer to topics discussed at lecture: so try to do some self study or brush up some notions.

1. Let's warm up. Set $T = 30$ and code an Metropolis-Hastings algorithm for inference on θ, where the likelihood is approximated using the bootstrap filter[1]. Use $N = 500$ particles.

   (a) Run Metropolis-Hastings for as many iterations you deemed necessary and, after removing the burnin draws, produce histograms of the posterior marginals for q and r.

   (b) compute posterior means and 95% posterior intervals for q and r. How did you obtain the latter? *(Not explicitly shown at lecture).*

2. Verify that using a "low" number of particles (e.g. $N = 20$) does somehow impact the quality of the inference. How? Why are the marginals so different from the case using $N = 500$ even though the algorithm is supposed to produce exact inference for any value of N? (this is something you learn in lecture 2, but you can also look at the paragraph below otherwise).

   There are two ways to reason about this fact: the first one is more intuitive and guided by running the code using the suggested setting. A further, more interesting, way is to look at section 1 (first three pages)

[1] the resulting MCMC algorithm is therefore going to constitute an instance of the pseudomarginal approach described in lecture 2. But no need to wait for lecture 2 to experiment.

of Sherlock, Thiery, Roberts and Rosenthal (2015). On the efficiency of pseudo-marginal random walk Metropolis algorithms. The Annals of Statistics 43(1), 238-275.

3. Now consider a generalization of the previous model:

$$\begin{cases} x_t = 0.5x_{t-1} + 25\frac{x_{t-1}}{(1+x_{t-1}^2)} + 8\cos(1.2(t-1)) + v_t, \\ y_t = cx_t^2 + e_t, \end{cases}$$

where c is a positive unknown constant. Conduct Bayesian inference for $\theta = (c, q, r)$. Set for c a fairly wide uniform prior with positive support, and keep the already set inverse-Gamma priors for (q, r). Use $(c, q, r) = (0.05, 0.1, 1)$ as "true values" to generate $T = 50$ observations. Set starting values for (c, q, r) fairly distant from the true data.

Hint: use care when specifying the value of the standard deviation of the proposal function (Gaussian random walk) for c. [2]

4. Consider the Ornstein-Uhlenbeck (OU) SDE model within the following state-space model (see slides in lecture 2):

$$\begin{aligned} dx_t &= -\beta(x_t - \alpha)dt + \sigma \cdot dB_t, \\ y_t &= x_t + e_t, \quad e_t \sim_{iid} N(0, 0.316^2) \end{aligned}$$

where

- $\alpha \in \mathbb{R}$ is the *stationary mean* of the process;
- $\beta > 0$ is the growth rate;
- $\sigma > 0$ diffusion coefficient (intensity of the intrinsic noise).

OU has known (Gaussian) transition densities. Here I write it explicitly for the evolution from x_t to $x_{t+\Delta}$ ($\Delta > 0$)

$$p(x_{t+\Delta}|x_t) = N\left(\alpha + (x_t - \alpha)e^{-\beta\Delta}, \frac{\sigma^2}{2\beta}(1 - \exp(-2\beta\Delta))\right).$$

However we wish to again use the bootstrap filter within Metropolis-Hastings, so for our purposes it is more useful to write *how* we simulate a path exactly (just a consequence of using the transition density above):

$$x_{t+\Delta} = \alpha + (x_t - \alpha)e^{-\beta\Delta} + \sqrt{\frac{\sigma^2}{2\beta}(1 - \exp(-2\beta\Delta))} \times \xi_{t+\Delta}$$

with $\xi_t \sim N(0, 1)$ iid. Consider the same settings (number of observations, priors, true parameters etc) as in the slides. Try to infer the data-generating model parameters.

[2]Extra thing for the curious ones: you may wonder if there exist adaptive strategies to automatically learn the "right" standard deviation for the proposal function to be used in Metropolis-Hastings. One of those is (simple to implement) the one in Haario et al. (2001) "An adaptive Metropolis algorithm", Bernoulli, 223-242 (equation (1) in the paper is all you need really).