Estimate Observed Information Matrix Via Particle Filtering Algorithm

Jiajie Kong[1]
Department of Statistics, University of California, Santa Cruz, CA, 95064, USA

Let us begin with the equation (65) in the paper

$$\hat{\boldsymbol{\mathcal{L}}}_T(\boldsymbol{\theta}, \boldsymbol{\eta}) = P(X_0 = x_0)\frac{1}{N}\sum_{i=1}^{N} w_T^i \tag{1}$$

Unfortunately, $\hat{\boldsymbol{\mathcal{L}}}_T(\boldsymbol{\theta}, \boldsymbol{\eta})$ is a random variable itself even condiftioned on fixed observed value $\{X_1, \cdots, X_T\}$. As far as we know noise is not differentiable, for example, multivariate brownian motion, gaussian process with matern kernel where smoothness parameter is less than 1, etc., in our case, parameters space can be treated as a "index" space. Now the key point is, I want to find a fixed number, instead of a r.v., whenever I get through the Particle Filtering Algorithm. So I conjure the following result maybe true:

$$P(X_0 = x_0)\frac{1}{N}\sum_{i=1}^{N} w_T^i \xrightarrow{p} C \quad \text{as } N \to \infty \tag{2}$$

we want N is large enough therefore they represent all the possible paths. Here is what my idea is, let T is the length of time, N is the number of sample path, then define

$$\boldsymbol{U} = \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,T-1} & u_{1,T} \\ u_{2,1} & u_{2,2} & \cdots & u_{2,T-1} & u_{2,T} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ u_{N-1,1} & u_{N-1,2} & \cdots & u_{N-1,T-1} & u_{N-1,T} \\ u_{N,1} & u_{N,2} & \cdots & u_{N-1,N-1} & u_{N-1,T} \end{bmatrix},$$

where $u_{ij} \sim \text{Unif}(0,1)$ for $\forall i, j$. Now the trick is, we need to use the same $\boldsymbol{U}$ whenever we apply the Particle Filtering Algorithmn, then the orginal random noise we need to sample will become:

[1]Email: jkong7@ucsc.edu

$$\begin{bmatrix} F^{-1}_{\hat{\sigma}_{1,1}}\left(u_{1,1}\right) & F^{-1}_{\hat{\sigma}_{1,2}}\left(u_{1,2}\right) & \cdots & F^{-1}_{\hat{\sigma}_{1,T-1}}\left(u_{1,T-1}\right) & F^{-1}_{\hat{\sigma}_{1,T}}\left(u_{1,T}\right) \\ F^{-1}_{\hat{\sigma}_{2,1}}\left(u_{2,1}\right) & F^{-1}_{\hat{\sigma}_{2,2}}\left(u_{2,2}\right) & \cdots & F^{-1}_{\hat{\sigma}_{T-1,1}}\left(u_{T-1,1}\right) & F^{-1}_{\hat{\sigma}_{T,1}}\left(u_{T,1}\right) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ F^{-1}_{\hat{\sigma}_{N-1,1}}\left(u_{N-1,1}\right) & F^{-1}_{\hat{\sigma}_{N-1,2}}\left(u_{N-1,2}\right) & \cdots & F^{-1}_{\hat{\sigma}_{N-1,T-1}}\left(u_{1,1}\right) & F^{-1}_{\hat{\sigma}_{N-1,T}}\left(u_{N-1,T}\right) \\ F^{-1}_{\hat{\sigma}_{N,1}}\left(u_{N,1}\right) & F^{-1}_{\hat{\sigma}_{N,2}}\left(u_{N,2}\right) & \cdots & F^{-1}_{\hat{\sigma}_{N,T-1}}\left(u_{N,T-1}\right) & F^{-1}_{\hat{\sigma}_{N,T}}\left(u_{N,T}\right) \end{bmatrix},$$

and $\hat{\sigma}_{ij}$ follow truncated normal distribution with different bounds. So basically we apply the copula idea here again. Notice that we don't need to care about the randomness of $\boldsymbol{U}$ here anymore since N is a large number choosen by us and will cover most possible outcome.

I try this trick on my code and the estimated $\hat{\boldsymbol{\theta}}_{MLE}$ is almost exactly the same as my original one, but now we have a fixed number.