Tutorial on Introduction to Sequential Monte Carlo methods

Tran Minh Ngoc
University of Sydney Business School

Vietnam National Conference on Probability and Statistics
Quy Nhon, August, 2025

Learning objectives

This tutorial is to introduce the Sequential Monte Carlo (SMC) method from a *practical* point of view. The main goal is that the participants can effectively use the method in their data analysis problem.

Materials

Lecture slides, code and data are available at
`https://github.com/VBayesLab/Tutorial-on-SMC`

Outline

Quick Introduction to Monte Carlo methods

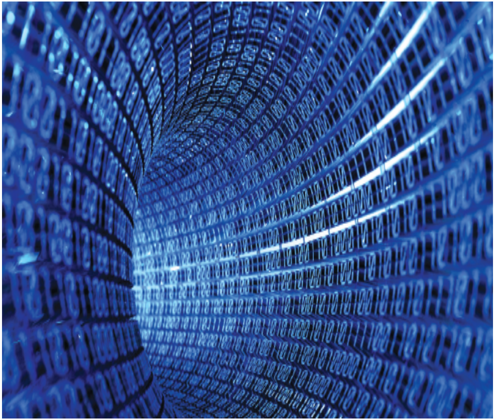
Motivating examples

Sequential Monte Carlo: Fixed Domain

Sequential Monte Carlo: Extended Domain

Monte Carlo methods

Monte Carlo simulation is at the heart of scientific computing[1]



[1] Picture credit: `https://sinews.siam.org/Details-Page/the-future-of-scientific-computation`

Monte Carlo methods

A basic problem in statistics is to compute an integral of the form

$$\mathcal{I} = \mathbb{E}[h(X)] = \int_{\mathcal{X}} h(x)\pi(x)dx$$

where $\pi(x)$ is a pdf on space $\mathcal{X}$, $X \sim \pi(x)$, and some function $h(x) : \mathcal{X} \mapsto \mathbb{R}$. Often, $\mathcal{X}$ is high dimensional.

E.g.

$$\mu_X = \mathbb{E}[X] = \int x\pi(x)dx, \quad \mathrm{Cov}(X) = \int (x - \mu_X)(x - \mu_X)^\top \pi(x)dx$$

$$\mathbb{P}(X \in A) = \int I_A(x)\pi(x)dx$$

where $I_A(x) = 1$ if and only if $x \in A$.

In most cases, we can't compute $\mathcal{I}$ analytically, have to use computers.

Monte Carlo problems

Monte Carlo methods are to deal with two main problems

P1 Generating samples from a probability distribution of interest with pdf $\pi(x)$.

P2 Estimating an integral of the form

$$\mathcal{I} = \int h(x)\pi(x)dx = \mathbb{E}_{X\sim\pi(x)}\big(h(X)\big)$$

for some function $h(x)$.

Problem 2 can be solved from Problem 1. Sometimes, it's more convenient and more efficient to solve Problem 2 directly.

Monte Carlo methods

Suppose that we are able to use a computer to generate

- i.i.d. samples $X_i \overset{iid}{\sim} \pi(x)$, $i = 1, \ldots, n$, or
- dependent, but ergodic Markov chain $\{X_i\}_{i \geq 1}$ with equilibrium distribution π.

Let

$$\widehat{\mathcal{I}}_n := \frac{1}{n} \sum_{i=1}^{n} h(X_i)$$

By LLN, $\widehat{\mathcal{I}}_n \xrightarrow{a.s.} \mathcal{I}$. For the iid case, $\mathbb{V}(\widehat{\mathcal{I}}_n) = \mathbb{V}(h(X))/n \longrightarrow 0$ as $n \to \infty$ regardless of the dimension of the integral.

Why MC methods work? Unlike numerical methods that spend all computational resources equally on the entire domain $\mathcal{X}$, Monte Carlo methods efficiently focus on regions with high π-density.

Monte Carlo methods

Example.

$$\mathcal{I} = \int_{\mathbb{R}} x^2 \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) dx = 1$$

Let try the following in Matlab

```
x = normrnd(0,1,n,1); Ihat = sum(x.*x)/n
```

n	100	1000	10000	1,000,000
$\widehat{\mathcal{I}}_n$	0.8098	0.9702	1.0057	1.0010
MC error	0.0569	0.0410	0.0312	0.0011

Note: *Reporting MC error is a "must" in any scientific reports that use MC.*

Monte Carlo methods

But wait... How can we use a computer to generate samples $X_i \sim \pi(x)$?

Note: there are two different uses of the notation $X \sim \pi(x)$. One means "distributed as" and the other means "sampled from".

Generating samples from a given probability distribution, or from a sequence of distributions, is the main focus of Monte Carlo methods.

This tutorial is about the Sequential Monte Carlo method, that generates samples from a sequence of distributions.

Outline

Example 1: Volatility modelling and forecasting

Let $y_{1:t} = \{y_1, ..., y_t\}$ be daily returns of a financial stock up to day t. Want to forecast the volatility $\sigma_{t+1}^2 = \mathbb{V}(y_{t+1}|y_{1:t})$.

GARCH model:

$$\begin{aligned} y_t &= \sigma_t \epsilon_t, \quad \epsilon_t \sim N(0,1) \\ \sigma_t^2 &= w + \alpha\sigma_{t-1}^2 + \beta y_{t-1}^2 \end{aligned}$$

The model parameter is $\theta = (w, \alpha, \beta)$. The posterior

$$\pi_t(\theta) = \frac{p(\theta)p(y_{1:t}|\theta)}{Z_t}, \; Z_t = \int p(\theta)p(y_{1:t}|\theta)d\theta$$

We want to sample from $\pi_t(\theta)$ sequentially. This provides a principled and convenient way to produce volatility forecast sequentially as data arrives

$$p(\sigma_{t+1}^2|y_{1:t}) = \int p(\sigma_{t+1}^2|\theta, y_t)\pi_t(\theta)d\theta, \; t = n+1, n+2, ...$$

We need some "long-enough" data $y_{1:n}$.

A bit about notations

In most cases, we only know the target density $\pi(x)$ up to a normalizing constant, i.e.,

$$\pi(x) = \frac{\gamma(x)}{Z}$$

where function $\gamma(x)$ is known but constant $Z = \int \gamma(x)dx$ is unknown.

Then, we often write

$$\pi(x) \propto \gamma(x)$$

to mean that we know $\pi(x)$ up to a constant.

E.g., the posterior distribution is often known up to a constant

$$\pi(\theta) = p(\theta|y) = \frac{1}{Z} \times \underbrace{p(\theta)}_{\text{prior}} \times \underbrace{p(y|\theta)}_{\text{likelihood}}, \text{ written as } \pi(\theta) \propto p(\theta)p(y|\theta)$$

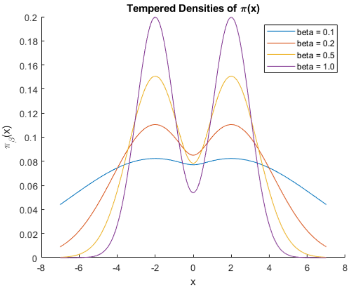
$Z = p(y) = \int p(\theta)p(y|\theta)d\theta$ is unknown, called marginal likelihood.

Example 2: Annealed importance sampling

- In many cases, it's challenging to sample from a target distribution $\pi(x)$: e.g., multimodal, non-standard shape
- A strategy is to design a sequence

$$\pi_t(x) \propto \pi(x)^{\beta_t}$$

  where $0 < \beta_1 < \cdots < \beta_T = 1$: sequence of temperatures.
- We first sample from π_1 (easy task as π_1 is very flat), then π_2, ..., until $\pi_T(x) = \pi(x)$.



Tempering annealed densities. If β_{t-1} is close to β_t, then π_{t-1} is close to π_t.

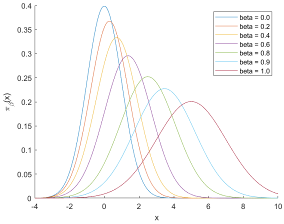
Example 3: Sequential sampling from bridged densities

- π_0: distribution that's easy to sample from; e.g., prior.
- $\pi(x)$: target distribution; e.g., the posteiror.
- Construct a sequence of "bridging" distributions from π_0 to π as follow

$$\pi_t(x) \propto \underbrace{\pi_0(x)}_{\text{easy}}{}^{1-\beta_t} \underbrace{\pi(x)}_{\text{difficult}}{}^{\beta_t},$$

$$0 = \beta_0 < \cdots < \beta_T = 1.$$

- We first sample from π_0 (easy to sampe from by definition), then π_1, ..., until $\pi_T(x) = \pi(x)$.



Bridging densities from $N(0,1)$ to $N(5,2)$.

In case $\pi(\theta)$ is a posterior $\pi_0(\theta) \propto p(\theta)p(y|\theta)$, with prior $\pi_0(\theta)$ and likelihood $p(y|\theta)$,

$$\pi_t(\theta) \propto \pi_0(\theta)p(y|\theta)^{\beta_t}$$

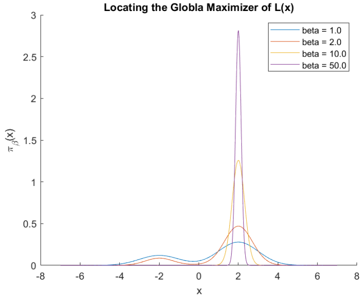
Often known as likelihood annealing method.

Example 4: Finding global maximizer

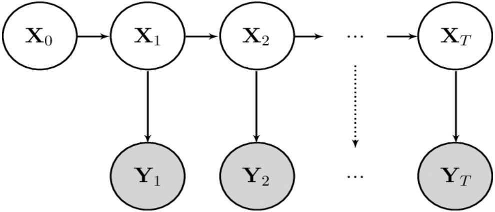
- $f(x)$: a non-negative objective function to maximize.
- construct a sequence of distributions as follow

$$\pi_t(x) \propto f(x)^{\beta_t}$$

  where $\beta_t \uparrow \infty$.
- for large β_t, $\pi_t(x)$ concentrates its mass at the global maximizer of $f(x)$.



Example 5: State-space models



$\{X_t\}_{t\geq 0}$: hidden/latent Markov process with

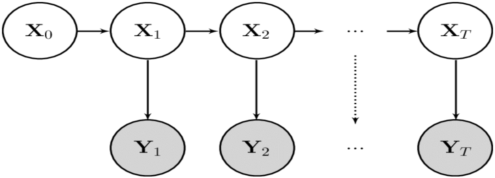
$$X_0 \sim \mu(x_0), \quad X_t|X_{t-1} = x_{t-1} \sim f(x_t|x_{t-1})$$

$\{Y_t\}_{t\geq 1}$: observed process, independent conditionally on $\{X_t\}_{t\geq 1}$:

$$Y_t|X_t = x_t \sim g(y_t|x_t)$$

Main task: Sample from $p(x_{1:t}|y_{1:t})$ for $t = 1, 2, \ldots$. Known as the particle filter problem.

Example 5: State-space models



There are abundant of state-space models, used in many scientific fields.

Stochastic volatility model: Let $\{Y_t\}_{t\geq 1}$ be financial asset returns

$$\begin{aligned} Y_t|X_t = x_t &\sim N\left(0, e^{x_t}\right) \\ X_t|X_{t-1} = x_{t-1} &\sim N(\mu + \phi x_{t-1}, \sigma^2). \end{aligned}$$

Here $\{X_t = \log \mathbb{V}(Y_t|X_t)\}_{t\geq 1}$ is the log-volatility process of interest. Financial risk management requires samples from X_1 (given y_1), then X_2 (given $y_{1:2}$), etc.

Two settings of SMC problems

Fixed domain: we want to approximate a sequence of distributions with the same domain

$$\pi_t(x), \quad x \in \mathcal{X}, \quad t = 1, 2, \ldots.$$

Examples 1-4 above belong to this setting.

Extended domain: we want to approximate the sequence of distributions with extended domain

$$\pi_t(x_{1:t}), \quad x_{1:t} \in \mathcal{X}^{\otimes t}, \quad t = 1, 2, \ldots.$$

Example 5, state-space models, belongs to this setting.

The two settings share many similarities in term of algorithmic design, but also have some substantial difference.

Outline

Sequential Monte Carlo: Fixed Domain

- Let $\{\pi_t\}_{t\geq 1}$ be a sequence of distributions defined on $\mathcal{X}$; each $\pi_t(x)$ is *known up to a normalizing constant*

$$\pi_t(x) = \underbrace{\frac{1}{Z_t}}_{\text{unknown}} \times \underbrace{\gamma_t(x)}_{\text{computable for every } x}$$

- Task: Sample from π_t and estimate Z_t *sequentially*: first sample from π_1 and estimate Z_1, then sample from π_2 and estimate Z_2, etc.

At each time t, we will use a set of weighted samples $\{W_t^{(i)}, X_t^{(i)}\}_{i=1}^M$, called particles, to approximate $\pi_t(x)$. That is, $\pi_t(x)$ is approximated by

$$\widehat{\pi}_t(x) = \sum_{i=1}^{M} W_t^{(i)} \delta_{X_t^{(i)}}(x)$$

where $0 \leq W_t^{(i)} \leq 1$, $\sum_i W_t^{(i)} = 1$. δ is the Dirac function.

SMC is based on Importance Sampling idea

Intuition: If the consecutive distributions π_t and π_{t-1} are "close" to each other for all t, then samples from π_{t-1} should be used to assist with sampling from π_t.

$$
\begin{aligned}
\int g(x)\pi_t(x)dx &= \int g(x)\gamma_t(x)dx \Big/ \int \gamma_t(x)dx \\
&= \int g(x)\frac{\gamma_t(x)}{\gamma_{t-1}(x)}\pi_{t-1}(x)dx \Big/ \int \frac{\gamma_t(x)}{\gamma_{t-1}(x)}\pi_{t-1}(x)dx \\
&= \int g(x)w_t(x)\pi_{t-1}(x)dx \Big/ \int w_t(x)\pi_{t-1}(x)dx \\
&= \int g(x)W_t(x)\pi_{t-1}(x)dx
\end{aligned}
$$

where $w_t(x) = \frac{\gamma_t(x)}{\gamma_{t-1}(x)}$ is unnormalized weight, and $W_t(x) = w_t(x) \Big/ \int w_t(z)\pi_{t-1}(z)dz$ is normalized weight.

Sequential Monte Carlo: Weighted Particles

Suppose at time $t-1$, we have a set of particles $\{X_{t-1}^{(i)}\}_{i=1}^{M}$ approximating $\pi_{t-1}(x)$.

As

$$\int g(x)\pi_t(x)dx = \int g(x)W_t(x)\pi_{t-1}(x)dx,$$

the weighted particles $\{W_t^{(i)}, X_{t-1}^{(i)}\}_{i=1}^{M}$ approximate $\pi_t(x)$ where

$$W_t^{(i)} = \frac{w_t\left(X_{t-1}^{(i)}\right)}{\sum_{i=1}^{M} w_t\left(X_{t-1}^{(i)}\right)}, \quad i = 1, ..., M$$

That is,

$$\widehat{\pi}_t(x) = \sum_{i=1}^{M} W_t^{(i)} \delta_{X_{t-1}^{(i)}}(x)$$

Sequential Monte Carlo: Resampling

The weighted particles $\{W_t^{(i)}, X_{t-1}^{(i)}\}_{i=1}^M$ approximate $\pi_t(x)$.

We now resample the particles $X_{t-1}^{(i)}$ with respect to their weights $W_t^{(i)}$ to get equally weighted particles:

- Each particle $X_{t-1}^{(i)}$ is copied $N_t^{(i)}$ times, with $\mathbb{E}(N_t^{(i)}) = MW_t^{(i)}$, $\sum_i N_t^{(i)} = M$.
- This is resampling *with* replacement.
- Many resampling methods can be used: multinomial resampling, stratified resampling, residual resampling.

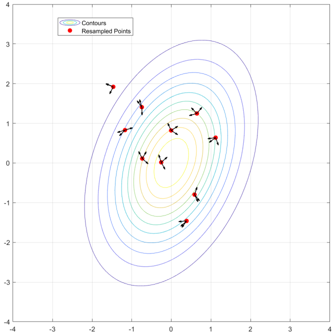
After resampling $\{W_t^{(i)}, X_{t-1}^{(i)}\}_{i=1}^M$, we obtain equally weighted particles $\{1/M, X_t^{(i)}\}_{i=1}^M$ approximating $\pi_t(x)$:

$$\widehat{\pi}_t(x) = \frac{1}{M}\sum_{i=1}^M \delta_{X_t^{(i)}}(x).$$

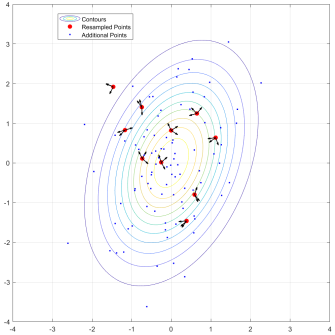
Sequential Monte Carlo: Depletion and Markov move

Resampling removes particles with low weights and replicates those with high weights. Might lead to only a few distinct particles - called depletion issue.

We need a Markov move step to "refresh" the particles, making them explore better the sample space.



Without Markov move



With Markov move

Sequential Monte Carlo: Depletion and Markov move

Markov move is often performed using the Metropolis-Hasting algorithm.

For each resampled particle $X_t^{(i)}$:

- Generate a proposal $X' \sim N(X_t^{(i)}, \Sigma_t)$, with Σ_t the sample covariance of the weighted particles $\{W_t^{(i)}, X_{t-1}^{(i)}\}_{i=1}^{M}$
- Set $X_t^{(i)} \leftarrow X'$ with the acceptance probability

$$\alpha = \min\left(1, \frac{\gamma_t(X')}{\gamma_t(X_t^{(i)})}\right),$$

otherwise keep $X_t^{(i)}$ unchanged.

$$\Sigma_t = \sum_i W_t^{(i)} X_{t-1}^{(i)} {X_{t-1}^{(i)}}^\top - \left(\sum_i W_t^{(i)} X_{t-1}^{(i)}\right)\left(\sum_i W_t^{(i)} X_{t-1}^{(i)}\right)^\top$$

Sequential Monte Carlo

In words, SMC moves a cloud of particles through the sequence of distributions $\{\pi_t(x)\}_{t=1}^{T}$ [2]. The cloud of particles at step t approximate $\pi_t(x)$.

The evolution of the particle cloud from one step to another consists of three steps: reweighting, resampling and moving.

[2] For simplicity, we assume that it is easy to sample from the initial distribution π_1

SMC for Fixed Domain: basic algorithm

Initialization: Generate M particles $X_1^{(i)} \sim \pi_1(x)$, $i = 1, ..., M$.

For $t = 2, ..., T$

- **Reweighting.** Compute normalized weights

$$w_t^{(i)} = \frac{\gamma_t(X_{t-1}^{(i)})}{\gamma_{t-1}(X_{t-1}^{(i)}))}, \quad W_t^{(i)} = w_t^{(i)} \Big/ \sum_j w_t^{(j)}$$

- **Resampling.** Resample weighted particles $\{W_t^{(i)}, X_{t-1}^{(i)}\}_{i=1}^M$ to get equally weighted particles $\{1/M, X_t^{(i)}\}_{i=1}^M$.
- **Markov move.** For $i = 1, ..., M$:
  1. Generate a proposal $X' \sim N(X_t^{(i)}, \Sigma_t)$
  2. Set $X_t^{(i)} \leftarrow X'$ with the acceptance probability

$$\alpha = \min\left(1, \frac{\gamma_t(X')}{\gamma_t(X_t^{(i)})}\right).$$

Note: for each particle, one often runs Markov step a few times, e.g., 5.

SMC for Fixed Domain: a bit on theory

Let $\{X_t^{(i)}\}_{i=1}^M$ be the set of particles approximating $\pi_t(x)$ at step t. The integral of interest $\mathbb{E}_{\pi_t}(g) = \int g(x)\pi_t(x)dx$ can be estimated by

$$\widehat{\mathbb{E}_{\pi_t}^{(M)}(g)} = \frac{1}{M}\sum_i g(X_t^{(i)})$$

It can be shown (Gilks and Berzuini, JRSSB 2001; Del Moral et al, JRSSB 2006) that

- $\widehat{\mathbb{E}_{\pi_t}^{(M)}(g)} \to \mathbb{E}_{\pi_t}(g)$ a.s. as $M \to \infty$
- $$\frac{\widehat{\mathbb{E}_{\pi_t}^{M}(g)} - \mathbb{E}_{\pi_t}(g)}{\sqrt{V_t(g)}} \Rightarrow N(0,1)$$

  $V_t(g) > 0$ has a complicated form (ignored here).

SMC for Fixed Domain: Estimating Z_t

SMC is well-known for its ability to estimate the normalizing constant Z_t.

$$\frac{Z_t}{Z_{t-1}} = \int w_t(x)\pi_{t-1}(x)dx$$

$$\widehat{\frac{Z_t}{Z_{t-1}}} = \frac{1}{M}\sum_i w_t\left(X_{t-1}^{(i)}\right)$$

If $Z_1 = 1$, then

$$Z_t = \frac{Z_2}{Z_1} \times \cdots \times \frac{Z_t}{Z_{t-1}}$$

$$\widehat{Z}_t = \prod_{s=1}^{t}\left(\frac{1}{M}\sum_i w_s\left(X_{s-1}^{(i)}\right)\right)$$

It can be shown that $\mathbb{E}(\widehat{Z}_t) = Z_t$.

Recap...

We have now covered the basic SMC algorithm.

Next, we will discuss in details some specific versions of SMC for various settings

- Likelihood Annealing SMC algorithm
- Data Annealing SMC algorithm

Likelihood Annealing SMC algorithm

Likelihood Annealing SMC is a special, but widely-used SMC algorithm: it moves a set of particles initially generated from the prior, to a set of particles approximating the posterior. It also provides an estimate of marginal likelihood - important for model comparison.

θ: model parameter

$p(\theta)$: prior distribution. Assume that we can sample from it and evaluate it.

$p(y|\theta)$: likelihood function.

Sequence of likelihood annealing distributions:

$$\pi_t(\theta) \propto p(\theta)p(y|\theta)^{a_t}$$

with annealing levels $0 = a_0 < a_1 < ... < a_T = 1$.

Note that: $\pi_0(\theta) = p(\theta)$ and $\pi_T(\theta) = p(\theta|y)$ the posterior.

Likelihood Annealing SMC algorithm

How to select T? In general we need a large T when the posterior is "weird" or high-dimensional. How to select a_t? Naive choice $a_t = t/T$ not always works well.

Guiding principle: selecting $\{a_t\}$ such that π_t is close enough to π_{t-1}. Specifically, we want the normalized importance weights

$$W_t(\theta) = w_t(\theta) \Big/ \int w_t(\theta)\pi_{t-1}(\theta)d\theta, \quad w_t(\theta) = p(y|\theta)^{a_t - a_{t-1}}$$

are of a high quality, i.e. having a small variance.

Let $\{\theta_{t-1}^{(i)}\}_{i=1}^M$ be M samples from $\pi_{t-1}(\theta)$. The quality of the weighted particles $\{W_t(\theta_{t-1}^{(i)}), \theta_{t-1}^{(i)}\}_{i=1}^M$, as an approximation of $\pi_t(\theta)$, is measured by Effective Sample Size

$$\mathrm{ESS}(a_t) = \frac{1}{\sum_{i=1}^M \left(W_t(\theta_{t-1}^{(i)})\right)^2}$$

- $0 < \mathrm{ESS} \leq M$. Higher ESS the better.

Likelihood Annealing SMC algorithm

Adaptive method for selecting the annealing levels $\{a_t\}$:

- Select an initial large T, e.g. $T = 10,000$.
- Let $\tilde{a}_i = (i/T)^3$, $i = 0, 1, ..., T$. Why cubic?
- Let $a_0 = \tilde{a}_0 = 0$.
- Select $a_1 = \min\{\tilde{a}_i : \tilde{a}_i > a_0 \text{ and } \mathrm{ESS}(\tilde{a}_i) < cM\}$, for some 0¡c¡1. That is, the next annealing level a_1 is the smallest $\tilde{a}_i > a_0$ such that ESS computed at $\tilde{a}_i$ is less than the threshold cM.
- c is a subjective choice. Common chocie $c = 0.8$.
- Select $a_2 = \min\{\tilde{a}_i : \tilde{a}_i > a_1 \text{ and } \mathrm{ESS}(\tilde{a}_i) < cM\}$. That is, the next annealing level a_2 is the smallest $\tilde{a}_i > a_1$ such that ESS computed at $\tilde{a}_i$ is less than the threshold cM.
- etc.

Likelihood Annealing SMC algorithm

Initialization: Sample $\theta_j \sim p(\theta)$ for $j = 1, ..., M$, $t \leftarrow 0, a_t \leftarrow 0$, log_llh $\leftarrow 0$.

While $a_t < 1$:

- $t \leftarrow t + 1$
- **Select** a_t **and reweighting:**
  - For each i such that $\tilde{a}_i > a_{t-1}$, compute

$$w_j = p(y|\theta_j)^{\tilde{a}_i - a_{t-1}}, \;\; W_j \propto w_j, \;\; \mathrm{ESS}_i = 1/\sum_{j=1}^{M} W_j^2$$

  - Increase i until $\mathrm{ESS}_i < cM$ for some $0 < c < 1$. Set $a_t \leftarrow \tilde{a}_i$.
- **Resampling** $\{W_j, \theta_j\}_{j=1}^M$ to obtain the new equally-weighted particles $\{\theta_j\}_{j=1}^M$.
- **Markov move**: For each $j = 1, ..., M$,
  1. Generate a proposal $\theta_j' \sim \mathcal{N}(\theta_j, \Sigma_t)$
  2. Set $\theta_j = \theta_j'$ with the probability $\min\left(1, \frac{p(y|\theta_j')^{a_t} p(\theta_j')}{p(y|\theta_j)^{a_t} p(\theta_j)}\right)$
- **Update log of marginal likelihood**: log_llh := log_llh + $\log\left(\sum_{j=1}^M w_j\right)$.

Example: Likelihood Annealing SMC for GARCH model

Let $\{y_t, t = 1, ...,\}$ be financial returns. Wish to model $\sigma_t^2 = \mathbb{V}(y_t | y_{1:t-1})$.

The GARCH model:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \sim N(0, 1)$$
$$\sigma_t^2 = w + \alpha \sigma_{t-1}^2 + \beta y_{t-1}^2, \quad t = 2, 3, ...$$

where $w > 0$, $\alpha > 0, \beta > 0$ and $\alpha + \beta < 1$. Let's parameterize $\alpha = \psi_1(1 - \psi_2)$ and $\beta = \psi_1 \psi_2$ with $0 < \psi_1, \psi_2 < 1$.

Use an inverse Gamma prior IG$(1, 1)$ for w and an uniform prior $U(0, 1)$ for ψ_1 and ψ_2.

Hence, the working model

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \sim N(0, 1)$$
$$\sigma_t^2 = w + \psi_1(1 - \psi_2)\sigma_{t-1}^2 + \psi_1 \psi_2 y_{t-1}^2, \quad t = 2, 3, ...$$

The model parameters $\theta = (w, \psi_1, \psi_2)$.

Example: Likelihood Annealing SMC for GARCH model

Data: SP500 weekly indexes from Jan 1988 to Nov 2018, leading to $N = 1612$ returns.

Use the first $y_{1:n=1000}$ as training data.

We want to sample from the posterior $p(\theta|y_{1:n})$ and estimate the marginal likelihood

$$p(y_{1:n}) = \int p(\theta)p(y_{1:n}|\theta)d\theta$$

Code running!

Left blank intentionally

Example: Likelihood Annealing SMC for GARCH model

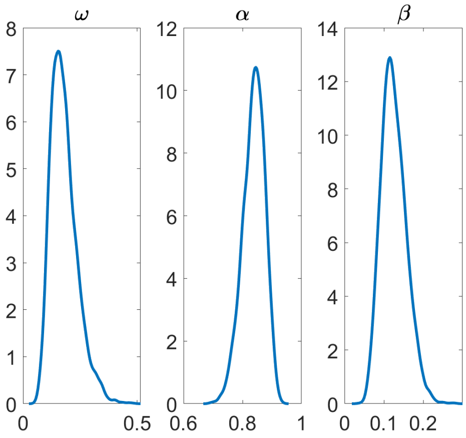


Figure: Posterior distributions of w, α and β. Estimated by Likelihood Annealing SMC algorithm.

Example: Likelihood Annealing SMC for LSTM-GARCH model

Nguyen et al. JAE 2024 propose the LSTM-GARCH model:

$$
\begin{aligned}
y_t &= \sigma_t \epsilon_t, \quad \epsilon_t \overset{iid}{\sim} t_\nu && t = 1, 2, ..., n \\
\sigma_t^2 &= \omega_t + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2 && t = 2, ..., n \\
\omega_t &= \beta_0 + \beta_1 h_t && t = 2, ..., n, h_1 = 0 \\
h_t &= g_t^o \times \text{sig}(C_t) && \text{LSTM cell output} \\
g_t^f &= \tanh(v_f x_t + w_f h_{t-1} + b_f) && \text{forget gate} \\
g_t^i &= \tanh(v_i x_t + w_i h_{t-1} + b_i) && \text{input gate} \\
x_t^d &= \tanh(v_d x_t + w_d h_{t-1} + b_d) && \text{data input} \\
g_t^o &= \text{sig}(v_o x_t + w_o h_{t-1} + b_o) && \text{output gate} \\
C_t &= g_t^f \times C_{t-1} + g_t^i \times x_t^d && \text{cell state}
\end{aligned}
$$

GARCH v.s. LSTM-GARCH model?

Model comparison and selection is an essential in statistical modelling!

- Likelihood Annealing SMC provides an efficient estimate of the marginal likelihood - key quantity for model comparison.
- For SP500 data:

	GARCH	LSTM-GARCH
log-llh estimate	-2077.95	-2056.10

Table: log of marginal likelihood estimate for GARCH and LSTM-GARCH models

Data Annealing SMC algorithm

Likelihood annealing SMC is suitable for in-sample analysis, as it approximates the posterior $p(\theta|y_{1:n})$ where $y_{1:n}$ denotes the training data.

For out-of-sample data-expanding forecasts where posterior of θ is updated once new data arrive, we use Data Annealing SMC: generating particles from

$$
\begin{aligned}
\pi_0(\theta) &= p(\theta|y_{1:n}) \\
\pi_1(\theta) &= p(\theta|y_{1:n+1}) \propto p(y_{1:n+1}|\theta)p(\theta) \propto \pi_0(\theta)p(y_{n+1}|\theta, y_{1:n}), \\
\ldots & \quad \ldots \\
\pi_t(\theta) &= p(\theta|y_{1:n+t}) \propto \pi_{t-1}(\theta)p(y_{n+t}|\theta, y_{1:n+t-1}), t = 2, 3, \ldots
\end{aligned}
$$

$$
\pi_t(\theta) \propto \underbrace{\pi_{t-1}(\theta)}_{\text{previous posterior}} \underbrace{p(y_{n+t}|\theta, y_{1:n+t-1})}_{\text{new information}}.
$$

Data Annealing SMC algorithm

Wish to approximate thre sequence of data-expanding distributions

$$\pi_t(\theta) \quad = \quad p(\theta|y_{1:n+t}) \propto \pi_{t-1}(\theta)p(y_{n+t}|\theta, y_{1:n+t-1}), t = 1, 2, ...$$

At time $t=0$, we already have a set of particles $\{W_j,\theta_j\}_{j=1}^M$ approximating $\pi_0(\theta)$ using likelihood annealing SMC.

At each time $t>0$, given a set of particles $\{W_j,\theta_j\}_{j=1}^M$ approximating $\pi_t(\theta)$:

- we can approximate the posterior predictive distribution of future data y_{n+t+1}

$$p(y_{n+t+1}|y_{1:n+t})=\int p(y_{n+t+1}|\theta,y_{1:n+t})\pi_t(\theta)d\theta$$

- when the data point y_{n+t+1} is available, we update π_{t+1} via

$$\pi_{t+1}(\theta)\propto\pi_t(\theta)p(y_{n+t+1}|\theta,y_{1:n+t}).$$

Fixed-data v.s. data-expanding forecast approaches

Fixed-data forecast approach computes the posterior predictive distribution of future data y_{n+t+1} as

$$p(y_{n+t+1}|y_{1:n}) = \int p(y_{n+t+1}|\theta, y_{1:n+t})\pi_0(\theta)d\theta.$$

Data-expanding forecast approach computes the posterior predictive distribution of future data y_{n+t+1} as

$$p(y_{n+t+1}|y_{1:n+t}) = \int p(y_{n+t+1}|\theta, y_{1:n+t})\pi_t(\theta)d\theta.$$

The latter takes into account the new information.

Data Annealing SMC algorithm

Given weighted particles $\{\theta_j, W_j\}_{j=1}^M$ approximating $\pi_0(\theta)$.
For $t = 0, 1, ...$

- **Forecasting:** Use the weighted particles $\{W_j, \theta_j\}_{j=1}^M$ for predicting y_{n+t+1}
- **Updating:** Given data y_{n+t+1}, approximate distribution π_{t+1}
  - compute weights $w_j = W_j p(y_{n+t+1}|y_{1:n+t}, \theta_j)$, $W_j \propto w_j, j = 1, ..., M$ and $\text{ESS} = \frac{1}{\sum_{j=1}^M (W_j)^2}$.
  - **if** ESS $< cM$ for some $0 < c < 1$, **then**
    - **Resampling** from $\{\theta_j, W_j\}_{j=1}^M$ to obtain the new equally-weighted particles $\{\theta_j, W_j = 1/N\}_{j=1}^M$.
    - **Markov move**: for each $j = 1, ..., M$
      - Generate a proposal $\theta_j' \sim N(\theta_j, \Sigma_t)$
      - Set $\theta_j = \theta_j'$ with the probability $\min\left(1, \frac{p(y_{1:n+t+1}|\theta_j')p(\theta_j')}{p(y_{1:n+t+1}|\theta_j)p(\theta_j)}\right)$.

Example: Data Annealing SMC for GARCH model

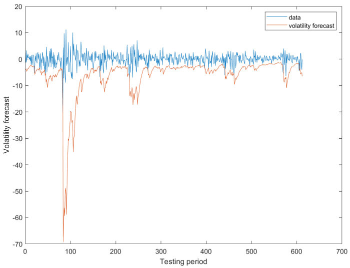
Data: SP500 weekly indexes from Jan 1988 to Nov 2018, leading to $N = 1612$ returns.

Use the first $y_{1:n=1000}$ as training data.

We want to construct one-step-ahead volatility forecast

$$\mathbb{V}(y_{n+t+1}|y_{1:n+1\theta}), \quad t = 0, 1, ...$$

Example: Data Annealing SMC for GARCH model



	PPS	Violate	Quantile Loss
Fixed-data forecast	2.2560	15	0.1120
Expanding-data forecast	2.1485	7	0.0965

Table: Forecast predictive metrics of fixed-data approach v.s. expanding-data approach.

Outline

Sequential Monte Carlo: Extended Domain

Not covered in this lecture!

Summary

We've covered the basic SMC algorithm and several specific versions of it.

We focused on the practical aspect of the method.

I hope this lecture equips data analysis practitioners with the tools and confidence to apply the method in practice.