Lecture 17 Sequential Monte Carlo Method *

Tiejun Li

1 Importance sampling with rejection

We have shown before that to evaluate the integral
\[ \int h(x)\pi(x)dx, \]
one can apply the importance sampling as the following steps:

- Draw \( X_1, \ldots, X_n \) i.i.d. from a distribution \( g(x) \).
- Calculate the importance weight
  \[ w_j = \frac{\pi(X_j)}{g(X_j)}, \quad \text{for } j = 1, 2, \ldots, n. \]
- Approximate the expectation by
  \[ \hat{\mu} = \frac{\sum_{i=1}^{n} w_i h(X_i)}{\sum_{i=1}^{n} w_i}. \quad (1) \]

The major advantage of using (1) instead of the unbiased estimate
\[ \bar{\mu} = \frac{1}{n} \sum_{i=1}^{n} w_i h(X_i). \]
is that in using the former, we need only the ratio \( \pi(x)/g(x) \) up to a multiplicative constant; whereas in the latter, the ratio needs to be known explicitly. Additionally, although inducing a small bias, (1) often has a smaller mean squared error than the unbiased one \( \bar{\mu} \).

When applying importance sampling, one often produces random samples with very small importance weights because of a less than ideal trial density. In this case we can combine the importance sampling with the rejection method introduced for generating random variables.

Suppose we have drawn samples \( X_1, \ldots, X_n \) from \( g(x) \). Let \( w_j = \pi(X_j)/g(X_j) \). We can conduct the following operation for any given threshold value \( c > 0 \).

**Algorithm 1** (Rejection control). The samples with small weight will be rejected with high probability in the rejection control process.

---

*School of Mathematical Sciences, Peking University, Beijing 100871, P.R. China*
• For \( j = 1, \ldots, n \), accept \( X_j \) with probability 
\[
r_j = \min \left\{ 1, \frac{w_j}{c} \right\}.
\]

• If the \( j \)-th sample \( X_j \) is accepted, its weight is updated to 
\[
w^*_j = q_c w_j / r_j = q_c \max\{w_j, c\},
\]
where 
\[
q_c = \int \min \left\{ 1, \frac{w_j}{c} \right\} g(x) dx.
\]
The constant \( q_c \) is maintained in \( w^*_j \) only for conceptual clarity instead of computational need since it will be canceled in the (1). The weight \( w_j \) should be updated because some samples are rejected which will introduce bias.

From the generating process of \( X_j \) with rejection, we know that actually the final samples \( X^*_j \) satisfy the distribution with density 
\[
g^*(x) = q_c^{-1} r(x) g(x) = q_c^{-1} \min \left\{ 1, \frac{w(x)}{c} \right\} g(x)
\]
where \( r(x) \in [0, 1] \). So from 
\[
\int h(x) \pi(x) dx = \int h(x) \frac{\pi(x)}{g^*(x)} g^*(x) dx
\]
the final weight should be 
\[
w^*(x) = \frac{\pi(x)}{g^*(x)} = \frac{\pi(x)}{g(x)} \frac{g(x)}{g^*(x)} = q_c \frac{w(x)}{r(x)},
\]
which is exactly the definition in the algorithm.

We note here that after applying rejection control we will typically have fewer than \( n \) samples. More samples can be drawn from \( g^*(x) \) to make up for the rejected examples.

2 Sequential importance sampling (SIS)

It is nontrivial to design a good trial distribution for doing importance sampling in high-dimensional problems. One of the most useful strategies in these problems is to build up the trial density sequentially. Suppose we can decompose \( \mathbf{x} \) as \( \mathbf{x} = (x_1, \ldots, x_d) \) where each of the \( x_j \) may be multidimensional. Then the trial density can be constructed as 
\[
g(\mathbf{x}) = g_1(x_1) g_2(x_2|x_1) \cdots g_d(x_d|x_1, \ldots, x_{d-1}).
\]
by which we hope to obtain some guidance from the target density while building up the trial density. Corresponding to the decomposition of \( \mathbf{x} \), if we rewrite the target density as 
\[
\pi(\mathbf{x}) = \pi(x_1) \pi(x_2|x_1) \cdots \pi(x_d|x_1, \ldots, x_{d-1}).
\]
where 
\[
\pi(x_t|x_{t-1}) = \frac{\pi(x_t)}{\pi(x_{t-1})}
\]
and
\[ \pi(x_t) = \int \pi(x_1, \ldots, x_d) dx_{t+1} \cdots dx_d. \] (2)

With this decomposition, the importance weight is
\[ w(x) = \frac{\pi(x_1)\pi(x_2|x_1)\cdots\pi(x_d|x_1, \ldots, x_{d-1})}{g_1(x_1)g_2(x_2|x_1)\cdots g_d(x_d|x_1, \ldots, x_{d-1})}, \]
and we can obtain it sequentially
\[ w_t(x_t) = w_{t-1}(x_{t-1}) \frac{\pi(x_t|x_{t-1})}{g_t(x_t|x_{t-1})}, \quad t = 1, \ldots, d \]
where \( w_0 = 1 \), the final \( w(x) = w_d(x) \). Though theoretically it is possible, the construction of \( \pi(x_t) \) from (2) is not feasible in practice!

Now suppose we can find a sequence of “auxiliary distributions” \( \pi_1(x_1), \pi_2(x_2), \ldots, \pi_d(x) \), so that \( \pi_t(x_t) \) is a reasonable approximation to \( \pi(x_1) \), for \( t = 1, \ldots, d-1 \) and \( \pi_d = \pi \). We want to emphasize that \( \pi_t \) are only required to be known up to a normalizing constant. The SIS method can then be defined as follows:

**SIS Step:**

- Draw \( X_t = x_t \) from \( g_t(x_t|x_{t-1}) \) and let \( x_t = (x_{t-1}, x_t) \).
- Compute
  \[ u_t = \frac{\pi_t(x_t)}{\pi_{t-1}(x_{t-1})g_t(x_t|x_{t-1})}, \]
  and let \( w_t = w_{t-1}u_t \).

In the SIS step, we call \( u_t \) an “incremental weight”. It is easy to show that \( x_t \) is properly weighted by \( w_t \) with respect to \( \pi_t \) provided that \( x_{t-1} \) is properly weighted by \( w_{t-1} \) with respect to \( \pi_{t-1} \). One reason for the sequential buildup of the trial density is that it breaks a difficult task into manageable pieces.

Some more concrete set up of the algorithm may be

- We can build \( g_t \) in light of \( \pi_t \). For example, one can choose (if possible)
  \[ g_t(x_t|x_{t-1}) = \pi_t(x_t|x_{t-1}) = \frac{\pi_t(x_t)}{\pi_t(x_{t-1})} \]
  then the incremental weight becomes
  \[ u_t = \frac{\pi_t(x_t|x_{t-1})}{\pi_{t-1}(x_{t-1})} \]
- When we observe that \( w_t \) is getting too small, we can choose to reject the sample halfway and restart again. In this way, we avoid wasting time on generating samples that are doomed to have little effect in the final estimation. However, the rejection control should be used to correct the bias.

Now we state the SIS with rejection control here. Suppose a sequence of “check points” \( 0 < t_1 < t_2 < \cdots < t_d \leq d \), and a sequence of threshold values \( c_1, \ldots, c_k \) are given in advance. The following procedure can be implemented:
3.1 Growing a polymer on the square lattice

At each check point \( t_j \), start \( RC(t_k) \) with threshold value \( c = c_j \). If the partial sample \( (x_1, \ldots, x_{t_j}) \) has a weight \( w_{t_j} \), then we accept this partial sample with probability \( \min\{1, w_{t_j}/c_j\} \) and if accepted, replace its weight by \( w_{t_j}^* = \max\{w_{t_j}, c_j\} \).

For each rejected partial example, restart from the beginning again and let it pass through all the check points at \( t_1, \ldots, t_k \), with threshold values \( c_1, \ldots, c_j \), respectively. If rejected in any middle check point, start again.

Since this method requires that each rejected sample be restarted from stage 0, it tends to be impractical when \( d \) is very large. Combining the RC operation with resampling will be described later.

3 Some applications of Sequential Monte Carlo method

The SIS-based MC have been invented independently in at least three main research areas. One is by simulating macromolecules in 1950’s, the other two are more recent: one is by statistical missing data problem (1990’s), and the other by nonlinear filtering problem (1990’s).

3.1 Growing a polymer on the square lattice

We consider generating a polymer with the self avoiding random walk (SAW) model. In this case,

\[
\pi(x) = \frac{1}{Z_N}
\]

where \( N \) is number of nodes of the polymer, and \( Z_N \) is just a normalization constant. The most naive way of simulating a SAW is to start a random walk at \((0,0)\), and at each step \( i \), the walker, for that he is not allowed to fall back to where it came from at step \( i - 1 \), chooses with equal probability on of the three allowed neighboring positions to go. If that position has already been visited earlier, the walker has to go back to \((0,0)\) and start a new chain again. For \( N = 20 \), the rate of obtaining a successful SAW is approximately 21.6\%, for \( N = 48 \), the rate is as low as 0.79\%.

![Self Avoiding random Walk](image)

Figure 1: The self avoiding random walk and a SAW on a 2D lattice space with \( N=33 \).

Without loss of generality, we assume that the SAW is always started at \((0,0)\) and pays its first visit to position \((1,0)\). Now suppose the random walker is located at \( x_t = (i,j) \), then the walker examines all of the
neighbors of \( x_t \). If all of the neighbors have been visited, the walk is terminated and restarted. Otherwise the walker choose one with equal probability. Mathematically it is

\[
P[x_{t+1} = (i', j')] | x_1, \ldots, x_t = \frac{1}{n_t}
\]

where \( n_t \) is the total number of such unoccupied neighbors.

It is easy to check that the SAWs produced by this growth method is not uniformly distributed. Hammersley, Morton and Rosenbluth noticed that a successfully produced SAW in this way needs to be assigned a weight computed as

\[
w(x) = n_1 \cdots n_{N-1}
\]

The above algorithm can be put into the SIS framework exactly. In the growth method, we actually set

\[
\pi_t(x_t) = \frac{1}{Z_t}, \quad g_t(x_t | x_{t-1}) = \frac{1}{n_{t-1}},
\]

and in this case,

\[
\pi_t(x_{t-1}) = \sum_{\text{all possible } x'} \pi_t(x_{t-1}, x_t) = \frac{n_{t-1}}{Z_t}
\]

So we have \( g_t(x_t | x_{t-1}) = \pi_t(x_t | x_{t-1}) \), and in this case the incremental weight \( u_t = n_{t-1} \).

A serious limitation of the growth method appears when one tries to simulate very large polymers, say with \( N = 250 \). In this case, the resampling method is needed.

### 3.2 Sequential imputation for statistical missing data problem

In a missing data problem, we partition \( y \) as \((y_{\text{obs}}, y_{\text{mis}})\), where only \( y_{\text{obs}} \) is observed and \( y_{\text{mis}} \) is called the missing data. Furthermore we assume \( y \sim f(y|\theta) \). Suppose \( y_{\text{obs}} \) and \( y_{\text{mis}} \) can each be further decomposed into \( n \) corresponding components so that

\[
y = (y_1, \ldots, y_n) = (y_{\text{obs},1}, y_{\text{mis},1}, \ldots, y_{\text{obs},n}, y_{\text{mis},n}) = (y_{\text{obs}}, y_{\text{mis}}),
\]

where \( y_t = (y_{\text{obs},t}, y_{\text{mis},t}) \) for \( t = 1, \ldots, n \) (for example the state-space model). Let us consider the Bayesian computation of the parameter \( \theta \).

The joint distribution

\[
f(\theta, y_{\text{mis}}, y_{\text{obs}}) = f(y_{\text{mis}}, y_{\text{obs}} | \theta) f(\theta)
\]

where \( f(\theta) \) is the prior distribution. And the posterior distribution can be written as

\[
f(\theta | y_{\text{obs}}) = f(\theta | y_{\text{mis}}, y_{\text{obs}}) f(y_{\text{mis}} | y_{\text{obs}}) dy_{\text{mis}}.
\]

If a random sample \( y_{\text{mis}}^1, \ldots, y_{\text{mis}}^m \) can be generated from \( f(y_{\text{mis}} | y_{\text{obs}}) \), one can approximate the posterior as

\[
f(\theta | y_{\text{obs}}) = \frac{1}{m} \sum_{j=1}^m f(\theta | y_{\text{mis}}^j, y_{\text{obs}}).
\]

Since generating from \( f(y_{\text{mis}} | y_{\text{obs}}) \) is typically difficult, we can implement the sequential imputation.
We start by drawing $y_{mis,1}$ from $f(y_{mis,1}|y_{obs,1})$ and computing $w_1 = f(y_{obs,1})$. Note that
\[
f(y_{mis,1}|y_{obs,1}) \propto \int f(y_{mis,1}, y_{obs,1}|\theta) f(\theta) d\theta
\]
This integration can be done analytically in some cases. For $t = 2, \ldots, n$, the following are done sequentially:

- Draw $y_{mis,t}$ from the conditional distribution
  \[
f(y_{mis,t}|y_{obs,t}, y_{mis,t-1}).
  \]
  Notice that each $y_{mis,t}$ is drawn conditional on the previously imputed missing components $y_{mis,t-1}$.

- Compute the weight recursively
  \[
w_t = w_{t-1} f(y_{obs,t}|y_{obs,t-1}, y_{mis,t-1}).
  \]

Finally we let
\[
w(y_{mis}) = w_n = f(y_{obs,1}) \prod_{t=2}^{n} f(y_{obs,t}|y_{obs,t-1}, y_{mis,t-1})
\]
When integrating out $\theta$ is analytically is not feasible, one can treat $\theta$ as an additional missing data, $y_{mis,\theta}$, and use the same method above.

Finally we can estimate the posterior distribution $f(\theta|y_{obs})$ by the weighted mixture
\[
\frac{1}{W} \sum_{j=1}^{m} w^j f(\theta|y_{obs}, y_{mis}^j)
\]
where $W = \sum_{j} w^j$.

Similar problems appear when $n$ is very large, the weight $w_n$ become very skewed. The resample method is needed.

### 3.3 Nonlinear filtering

In this section we consider the state-space model (a kind of Graphical model) with the equations:

state equation: $x_t \sim q_t(\cdot|x_{t-1}, \theta)$

observation equation: $y_t \sim f_t(\cdot|x_t, \phi)$

From the optimal prediction theory, the task is to estimate the condition mean of $x_t$ of the a posteriori distribution. That is,
\[
\hat{x}_t = \mathbb{E}(x_t|y_1, \ldots, y_t) = \int x_t \pi_t(x_t) dx_t
\]
where
\[
\pi_t(x_t) = P(x_t|y_1, \ldots, y_t) \propto \int q_t(x_t|x_{t-1}) f_t(y_t|x_t) \pi_{t-1}(x_{t-1}) dx_{t-1}
\]
is the posterior distribution.
If $f_t, q_t$ are linear Gaussian conditional distribution, that corresponds to the Kalman filtering problem, if $x_t$ only takes finite states, that corresponds to the HMM model. Here for the general filtering problem, the scheme is called the sequential importance-resampling (SIR) (Rubin 1987).

Suppose at time $t$ that we have a random sample $\{x_{1:t}^1, \ldots, x_{1:t}^m\}$ of the state variable which follow approximately the current posterior distribution $\pi_t(x_t) = P(x_t | y_1, \ldots, y_t)$. Gordon (1993) suggested the following updating procedure when $y_{t+1}$ is observed:

- Draw $x_{t+1}^j$ from the state equation $q_t(x_{t+1} | x_t^j)$, $j = 1, \ldots, m$.
- Weight each draw by $w^j \propto f_t(y_{t+1} | x_{t+1}^j)$.
- Resample from $\{x_{t+1}^1, \ldots, x_{t+1}^m\}$ with probability proportional to $w^j$ to produce a random sample $\{x_{t+1}^1, \ldots, x_{t+1}^m\}$ for time $t+1$.

It is easy to show that if the $x_{t+1}^m$ follow the current posterior distribution $\pi_t(x_t)$ and if $m$ is large enough, then the new random sample $\{x_{t+1}^1, \ldots, x_{t+1}^m\}$ follows the updated posterior distribution $\pi_{t+1}(x_{t+1})$ approximately.

Under the SIS framework, we see that the current target distribution $\pi_t(\cdot)$ satisfies the recursion

$$\pi_t(x_t) \propto f_t(y_t | x_t) q_t(x_t | x_{t-1}) \pi_{t-1}(x_{t-1})$$

A special feature of the state-space model is that the state variable $x_t$ possess a Markovian structure. This feature makes it possible for one to consider only the marginal posterior distribution $\pi_t(x_t)$, instead of the joint distribution $\pi_t(x_t)$.

References