Time Series Analysis

7. Particle filters

Andrew Lesniewski

Baruch College
New York

Spring 2017

Outline

Bayesian estimation

- Estimation of complex time series models requires evaluation of complex expected values, often expressed as high dimensional, analytically intractable integrals. This is best done within the Bayesian approach to inference, which utilizes Markov chain Monte Carlo methods.
- *Particle filters*, a.k.a. *sequential Monte Carlo* (SMC) methods, provide an alternative to MCMC for calculating such integrals.
- In this approach, a continuous PDF is approximated by a discrete PDF made of weighted outcomes called *particles*.
- Particle filter algorithms are formulated recursively, very much in the spirit of the Kalman filter discussed in Lecture Notes #5. They are also a far reaching generalizations of the Kalman filter to non-linear, non-Gaussian state space models.
- All the probability distributions in the following depend on some parameters θ. In order to streamline the notation, we will suppress θ from all the formulas.

Non-linear state space models

- Broadly speaking, the objective of a particle filter is to estimate the posterior PDF of the (unobserved) state variables given a time series of observation variables.
- The particle filter is designed for a general clss of state space models, with continuous or discrete state variables:

$$\begin{array}{ccccccccc} X_0 & \rightarrow & X_1 & \rightarrow & \dots & \rightarrow & X_t & \rightarrow & \dots \\ & & \downarrow & & \downarrow & & \downarrow & & \dots \\ & & Y_1 & & \dots & & Y_t & & \dots \end{array} \tag{1}$$

  Such models include linear and non-linear Kalman filters, hidden Markov models, stochastic volatility models, etc.
- As discussed in Lecture Notes #5, the filtering problem is to estimate *sequentially* the values of the unobserved states X_t, given the values of the observation process $Y_0, \cdots, Y_t$, for any time step t.

Non-linear state space models

- All Bayesian inference about X_t is encoded in the posterior PDF $p(X_t \mid Y_{1:t})$. The particle filter methodology provides an approximation of these conditional probabilities using the empirical measure associated with a sampling algorithm.
- In contrast, the MCMC approach models the full posterior PDF $p(X_{0:t} \mid Y_{1:t})$.
- Particle methods usually assume X_t and the observations Y_t can be modeled in the following form. $X_0, X_1, \cdots$, is a Markov process on $\mathbb{R}^n$ (for some $n \geq 1$) that evolves according to the transition probability density $p(X_t \mid X_{t-1})$:

$$X_t \mid X_{t-1} \sim p(X_t \mid X_{t-1}). \tag{2}$$

On the other hand, Y_t depends only on the value of the state variable X_t:

$$Y_t \sim p(Y_t \mid X_t). \tag{3}$$

- Usually, these two relations are stated in explicit functional form:

$$\begin{aligned} X_t &= F(X_{t-1}, \varepsilon_t), \\ Y_t &= H(X_t, \eta_t), \end{aligned} \tag{4}$$

where ε_t and η_t are disturbances.

Stochastic volatility model

- An example of a non-linear state space model is the stochastic volatility model

$$\begin{aligned} X_{t+1} &= a + X_t + \varepsilon_{t+1}, \\ Y_t &= \exp(X_t)\eta_t, \end{aligned} \tag{5}$$

  where $a \in \mathbb{R}$, $\varepsilon \sim N(0, \alpha^2)$, and $\eta_t \sim N(0, 1)$.
- This model can be viewed as a discretized varsion of a continous time stochastic volatility model such as SABR. See related png file

Joint smoothing distribution

- Distribution properties of the state variable can be captured by the *joint smoothing distribution*, which is defined as

X0 is given

$$p(X_{0:t} \mid Y_{1:t}) = \frac{p(X_{0:t}, Y_{1:t})}{p(Y_{1:t})}. \tag{6}$$

- We derive the following recursion relation for the joint smoothing distribution:

Can be used for new upcoming information, recursively, no need to run the whole process from beginning everytime

See related png file with details steps

$$\begin{aligned} p(X_{0:t} \mid Y_{1:t}) &= \frac{p(Y_t \mid X_{0:t}, Y_{1:t-1}) p(X_{0:t}, Y_{1:t-1})}{p(Y_t, Y_{1:t-1})} \\ &= \frac{p(Y_t \mid X_{0:t}, Y_{1:t-1}) p(X_t \mid X_{0:t-1}, Y_{1:t-1})}{p(Y_t \mid Y_{1:t-1})} p(X_{0:t-1} \mid Y_{1:t-1}) \qquad (7) \\ &= \frac{p(Y_t \mid X_t) p(X_t \mid X_{t-1})}{p(Y_t \mid Y_{1:t-1})} p(X_{0:t-1} \mid Y_{1:t-1}). \end{aligned}$$

- In the particle filter algorithm, this recursion will be approximated by numerically tractable expressions.

Filtering recursion

- An alternative to working directly with the joint smoothing distribution is to find recursive relations for the one-period predictive and filtering distributions. This is analogous to the approach we took when deriving the Kalman filter.
- Assume that the initial distribution $p(X_0)$ is known. Then the one-period prediction distribution is given by

$$p(X_t \mid Y_{1:t-1}) = \int p(X_t \mid x_{t-1}) p(x_{t-1} \mid Y_{1:t-1}) dx_{t-1}. \tag{8}$$

- The filtering distribution is calculated based on the arrival of Y_t.
- Namely, applying Bayes' rule, and the fact that Y_t depends on X_t only,

$$\begin{aligned} p(X_t \mid Y_{1:t}) &= \frac{p(Y_t, X_t \mid Y_{1:t-1})}{p(Y_t \mid Y_{1:t-1})} \\ &= \frac{p(Y_t \mid X_t, Y_{1:t-1}) p(X_t \mid Y_{1:t-1})}{\int p(Y_t \mid x_t) p(x_t \mid Y_{1:t-1}) dx_t} \\ &= \frac{p(Y_t \mid X_t) p(X_t \mid Y_{1:t-1})}{\int p(Y_t \mid x_t) p(x_t \mid Y_{1:t-1}) dx_t}. \end{aligned} \tag{9}$$

Filtering recursion

- The difficulty with this recursion is clear: there is a complicated integral in the denominator, which cannot in general be calculated in closed form.
- In some special cases this can be done. For example, in the case of a linear Gaussian state space model, this integral is Gaussian and can be calculated (see Lecture Notes #5). The recursion above leads then to the Kalman filter.
- Instead of trying to evaluate the integral numerically, we will develop a Monte Carlo based approach for approximately solving recursions (8) and (9).

Importance sampling

- Suppose we are faced with Monte Carlo evaluation of the expected value

$$\mathrm{E}(f) = \int f(x_{0:t}) p(x_{0:t} \mid Y_{1:t}) dx_{0:t}, \tag{10}$$

where the the density $p(x_{0:t} \mid Y_{1:t})$ may be hard to simulate from.

- We proceed as follows. Choose a proposal distribution $g(X_{0:t} \mid Y_{0:t})$, and write

$$\mathrm{E}(f) = \int f(x_{0:t}) \frac{p(x_{0:t} \mid Y_{1:t})}{g(x_{0:t} \mid Y_{1:t})} g(x_{0:t} \mid Y_{1:t}) dx_{0:t}. \tag{11}$$

- Next, draw N samples of paths $x_{0:t}^1, \ldots, x_{0:t}^N$ from the proposal distribution, and assign to each of them a weight proportional to the ratio of the target and proposal distributions:

$$w_t^j \propto \frac{p(x_{0:t}^j \mid Y_{1:t})}{g(x_{0:t}^j \mid Y_{1:t})}. \tag{12}$$

Importance sampling

- Given the sample, we define

$$\widehat{\mathrm{E}}_N(f) = \sum_{j=1}^{N} \widehat{w}_t^j \, f(x_{0:t}^j), \tag{13}$$

where the *importance weights* $\widehat{w}_t^j$, $j = 1, \ldots, N$, are defined by

$$\widehat{w}_t^j = \frac{w_t^j}{\sum_{j=1}^{N} w_t^j} \, . \tag{14}$$

- The efficiency of this method depends essentially on how closely the proposal distribution $g(X_{0:t} \mid Y_{0:t})$ matches the target distribution. One could, for example, settle on a parametric distribution such as Gaussian and fine tune its parameters by minimizing its Kullback-Leibler divergence from $p(x_{0:t} \mid Y_{1:t})$.

Sequential importance sampling

- Another serious limitation of IS is that it is computationally very expensive, and that this cost increases with t.
- To mitigate it, the method of *sequential importance sampling* (SIS) has been developed. This is a version of IS where samples $x_{0:t}$ are simulated from a sequence of conditional distributions rather than a joint proposal distribution.
- The proposal distribution is factored into two pieces:

$$g(X_{0:t} \mid Y_{1:t}) = g(X_t \mid X_{0:t-1}, Y_{1:t}) g(X_{0:t-1} \mid Y_{1:t-1}). \tag{15}$$

- The second factor, $g(X_{0:t-1} \mid Y_{1:t-1})$, is the empirical distribution built out of the particles that have already been generated in the previous steps. A new set of samples $x_t^1, \ldots, x_t^N$ is drawn from the first factor $g(X_t \mid X_{0:t-1}, Y_{1:t})$.
- We then append the newly simulated values $x_t^1, \ldots, x_t^N$ to the simulated paths $x_{0:t-1}^1, \ldots, x_{0:t-1}^N$ of length t. We thus obtain simulated paths $x_{0:t}^1, \ldots, x_{0:t}^N$ of length $t+1$.

Sequential importance sampling

- The weights (12) are then given by

$$\begin{aligned} w_t &= \frac{p(Y_t \mid X_t)p(X_t \mid X_{t-1})p(X_{0:t-1} \mid Y_{1:t-1})}{p(Y_t \mid Y_{1:t-1})g(X_t \mid X_{0:t-1}, Y_{1:t})g(X_{0:t-1} \mid Y_{1:t-1})} \\ &\propto \frac{p(Y_t \mid X_t)p(X_t)}{g(X_{0:t-1} \mid Y_{1:t-1})} w_{t-1} \\ &\propto \widetilde{w}_t w_{t-1} \end{aligned} \tag{16}$$

  where $\widetilde{w}_t$ is defined by

$$\widetilde{w}_t = \frac{p(Y_t \mid X_t)p(X_t \mid X_{t-1})}{g(X_t \mid X_{0:t-1}, Y_{1:t})} . \tag{17}$$

- The densities $p(Y_t \mid X_t)$ and $p(X_t \mid X_{t-1})$ are determined by the state and observation equations (4). The only quantity that needs to be computed at each iteration is the ratio of weights $\widetilde{w}_t$.

Sequential importance sampling

- As a result of each iteration, SIS produces N Monte Carlo paths $x_{0:t}^1, \ldots, x_{0:t}^N$ along with importance weights $w_t^1, \ldots, w_t^N$. These paths are referred to as *particle*.
- The joint smoothing PDF is estimated as follows:

$$\widehat{p}(X_{0:t} \mid Y_{1:t}) = \sum_{j=1}^{N} \widehat{w}_t^j \, \delta(X_{0:t} - x_{0:t}^j), \qquad (18)$$
$$\widehat{w}_t^j = \frac{w_t^j}{\sum_{i=1}^{N} w_t^i},$$

and so

$$\widehat{\mathrm{E}}(f \mid Y_{1:t}) = \sum_{j=1}^{N} \widehat{w}_t^j \, f(x_{0:t}^j). \qquad (19)$$

- The estimated contribution to the likelihood function at time t is equal to

$$\widehat{p}(Y_t \mid Y_{1:t-1}) = \sum_{j=1}^{N} \widetilde{w}_t^j \, w_{t-1}^j. \qquad (20)$$

Sequential importance sampling with resampling

- SIS suffers from the *weight degeneracy* problem.
- While it does not recalculate the entire expression for the importance weights at each step, their variance increases exponentially. As the number of time steps increases, all the probability density gets eventually allocated to a single particle.
- That particle's normalized weight converges to one, while the normalized weights of the other partciles converge to zero, and the SIS estimator becomes a function of a single sample.
- A remedy to this problem is *resampling*. In this approach, a new population of particles is generated by sampling from the existing population:
  - (i) The probability of selecting a particle is proportional to its normalized importance weight.
  - (ii) Once the resampled particles are selected, their weights set equal (to $1/N$). This prevents the weights from degenerating as in SIS.

Similar as VaR (value at risk)

Sequential importance sampling with resampling

- Initialize the filter: draw N samples $x_0^j \sim g(X_0)$ and define the weights

$$w_0^j = \frac{p(x_0^j)}{g(x_0^j)} .$$

- For $t = 0, 1, \ldots, T$:
  - (i) Generate N samples $x_0^j \sim g(X_t \mid x_{0:t-1}^j)$ and compute the importance weights

$$w_t^j \propto \frac{p(Y_t \mid x_t^j)\, p(x_t^j \mid x_{t-1}^j)}{g(x_t^j \mid x_{t-1}^j, y_t)}\, w_{t-1}^j .$$

  - (ii) Normalize the importance weights:

$$\widehat{w}_t^j = \frac{w_t^j}{\sum_{j=1}^N w_t^j} . \tag{21}$$

  - (iii) Resample N particles with probabilities $\widehat{w}_t^1, \ldots, \widehat{w}_t^N$, and define $w_t^j = 1/N$.

Sequential importance sampling with resampling

- After every iteration, once the particles have been generated, quantities of interest can be estimated.
- The joint smoothing PDF at time t is estimated as follows:

$$\widehat{p}_N(X_{0:t} \mid Y_{1:t}) = \sum_{j=1}^{N} \widehat{w}_t^j \delta(X_{0:t} - x_{0:t}^j), \tag{22}$$

where the weights are given by (21).

- FThe estimate of the expected value of a function $f(X_{0:t})$ of the path $X_{0:t}$ is given by

$$\widehat{\mathrm{E}}_N(f \mid Y_{1:t}) = \sum_{j=1}^{N} \widehat{w}_t^j \delta(X_{0:t} - x_{0:t}^j). \tag{23}$$

- The contribution to the likelihood function at time t is estimated as follows:

$$\begin{aligned} \widehat{p}_N(Y_t \mid Y_{1:t-1}) &\approx \int p(Y_t \mid x_t) p(x_t \mid Y_{1:t-1}) dx_t \\ &\approx \frac{1}{N} \sum_{j=1}^{N} \widetilde{w}_t^j. \end{aligned} \tag{24}$$

Bootstrap filter

- The simplest choice of the proposal distribution is

$$g(X_t \mid X_{t-1}, Y_t) = p(X_t \mid X_{t-1}). \tag{25}$$

  This choice is called the *prior kernel*, and the corresponding particle filter is called the *bootstrap filter*.
- The bootsstrap filter resamples by setting the incremental weight ratios equal to $\widetilde{w}_t = p(Y_t \mid X_t)$.
- The prior kernel is an example of a *blind proposal*: it does not use the current observation Y_t.
- Despite this, the bootstrap filter performs well in a number of situations.
- Another popular version is the *auxiliary particle filter*, see [1] and [2].

Andrew: Bayesian is useful and the future

References


Creal, D.: A survey of sequential Monte Carlo methods for economics and finance, *Economic Reviews*, textbf31, 245 - 296 (2012).


Durbin, J., and Koopman, S. J.: *Time Series Analysis by State Space Methods*, Oxford University Press (2011).