STA 4103/5107 Computational Methods in Statistics II

Department of Statistics
Florida State University

Class 13
February 21, 2017

Next Five Classes

- Thursday, 02/23:
  Point Process Filter
- Tuesday, 02/28:
  Review for the Midterm
- Thursday, 03/02:
  No Class
- Tuesday, 03/07:
  Midterm Presentation, Group I
- Thursday, 03/09:
  Midterm Presentation, Group II

7.4 Sequential Monte Carlo Methods

Nonliniear Filtering

- In many problems it is not possible to assume linearity of system evolution and observation maps.
- Such problems require tools to handle nonlinearity of the underlying systems.
- An important consequence of nonlinear models is that the posterior density at any time may not be Gaussian, and hence its representation using a mean and a covariance is no longer sufficient.
- In such cases one resorts to Monte Carlo idea and generates a large number of samples from the posterior at every time t.

Recursive Estimation

- Those samples are then used to estimate system state, the estimation error, or any other parameter of interest.

- Since the posterior changes at every observation time, one would have to generate samples at every time.

- However, using the relationship between evolving posteriors, one seeks an efficient algorithm for generating samples from posterior at time $t + 1$, using samples from the posterior at time t, in a recursive fashion.

- In this section, we describe the method of **sequential Monte Carlo** to accomplish this task.

Goal

- Let $f(x_t|y_1, \ldots, y_t)$ be the posterior density function at time t, and let x_t^i be a sample from this posterior. Define the sample set $S_t = \{x_t^1, x_t^2, \ldots, x_t^n\}$.

- Our goal is to use elements of S_t, and the new observation y_{t+1} and the system-observation models, to generate the sample set S_{t+1}.

- This task is performed in two steps (analogous to Kalman Filtering).

- The first step generates prediction and the second step updates, or corrects, these predictions using the data y_{t+1}.

Prediction Step

- **1. Prediction Step:** This step uses samples from the posterior density at time t to generate samples from the prediction density $f(x_{t+1}|y_1, \ldots, y_t)$. That is,

$$f(x_{t+1} \mid y_1,\ldots,y_t) = \int_{x_t} f(x_{t+1} \mid x_t) f(x_t \mid y_1,\ldots,y_t)\, dx_t$$

- Since x_t^i is a sample from $f(x_t|y_1, \ldots, y_t)$, one can use a sample from the conditional density

$$f(x_{t+1} \mid x_t^i)$$

- Let $\tilde{x}_{t+1}^i \sim f(x_{t+1} \mid x_t^i)$. Then

$$\tilde{x}_{t+1}^i \sim f(x_{t+1} \mid y_1, y_2, \ldots, y_t)$$

Prediction Set

- We will call the set

$$\tilde{S}_{t+1} = \{\tilde{x}^1_{t+1}, \tilde{x}^2_{t+1}, \ldots, \tilde{x}^n_{t+1}\},$$

  as the **prediction set**.

- This one-step-prior comes from a state equation involving an update function $F(\cdot)$ and a random term.

- In this situation, sampling from this prior is straightforward, according to:

$$\tilde{x}^i_{t+1} = F(x^i_t) + w^i_t,$$

  where w^i_t is a random sample from its given density function. In examples where w_t is multivariate normal this simulation is rather simple.

Update Step

- **2. Update Step:** This step uses the prediction set to generate samples from the posterior density $f(x_{t+1}|y_1, \ldots, y_{t+1})$.

- We are interested in estimating a parameter

$$\theta_{t+1} = \int g(x_{t+1}) f(x_{t+1} \mid y_1, \ldots, y_{t+1}) dx_{t+1},$$

for a given function g. In case of minimum mean square estimation, the function is g is identity and we are interested in estimating the posterior mean as an estimate of the state x_{t+1}.

- The function g is always known beforehand.

- These two goals are accomplished using **importance sampling** and **resampling** as described next.

Importance Sampling

- Using ideas from importance sampling we can rewrite the definition of θ_{t+1} as follows:

$$\theta_{t+1} = \int g(x_{t+1}) f(x_{t+1} \mid y_1, \ldots, y_{t+1}) dx_{t+1}$$

$$= \int g(x_{t+1}) \frac{f(y_{t+1} \mid x_{t+1}) f(x_{t+1} \mid y_1, \ldots, y_t)}{f(y_{t+1} \mid y_1, \ldots, y_t)} dx_{t+1}$$

$$= \int \frac{g(x_{t+1}) f(y_{t+1} \mid x_{t+1})}{f(y_{t+1} \mid y_1, \ldots, y_t)} f(x_{t+1} \mid y_1, \ldots, y_t) dx_{t+1}$$

- If we have samples from $f(x_{t+1}|y_1, \ldots, y_t)$, we can use them to estimate θ_{t+1}.

- However, notice that the quantity $f(y_{t+1}|y_1, \ldots, y_t)$ is still unknown.

Sum of Weights

- We have to estimate this quantity. This is done by considering:

$$f(y_{t+1} \mid y_1, y_2, \ldots, y_t) = \int f(y_{t+1} \mid x_{t+1}) f(x_{t+1} \mid y_1, y_2, \ldots, y_t) dx_{t+1}$$

- Once again, since we have samples from $f(x_{t+1}|y_1, \ldots, y_t)$, we can use them, and the likelihood function $f(y_{t+1}|x_{t+1})$ to estimate the left side.

- Define the weights

$$w_{t+1}^i = f(y_{t+1} \mid \tilde{x}_{t+1}^i),$$

and estimate $f(y_{t+1}|y_1, \ldots, y_t)$ using

$$\frac{1}{n} \sum_{i=1}^{n} w_{t+1}^i.$$

Estimation

- Using Monte-Carlo Method,

$$\hat{\theta}_{t+1} = \frac{1}{n} \sum_{i=1}^{n} \frac{g(\tilde{x}_{t+1}^{i}) w_{t+1}^{i}}{\frac{1}{n} \sum_{j=1}^{n} w_{t+1}^{j}}.$$

- Define the normalized weights

$$\tilde{w}_{t+1}^{i} = \frac{w_{t+1}^{i}}{\sum_{j=1}^{n} w_{t+1}^{j}},$$

we can restate the estimator of θ_{t+1} as

$$\hat{\theta}_{t+1} = \sum_{i=1}^{n} g(\tilde{x}_{t+1}^{i}) \tilde{w}_{t+1}^{i}.$$

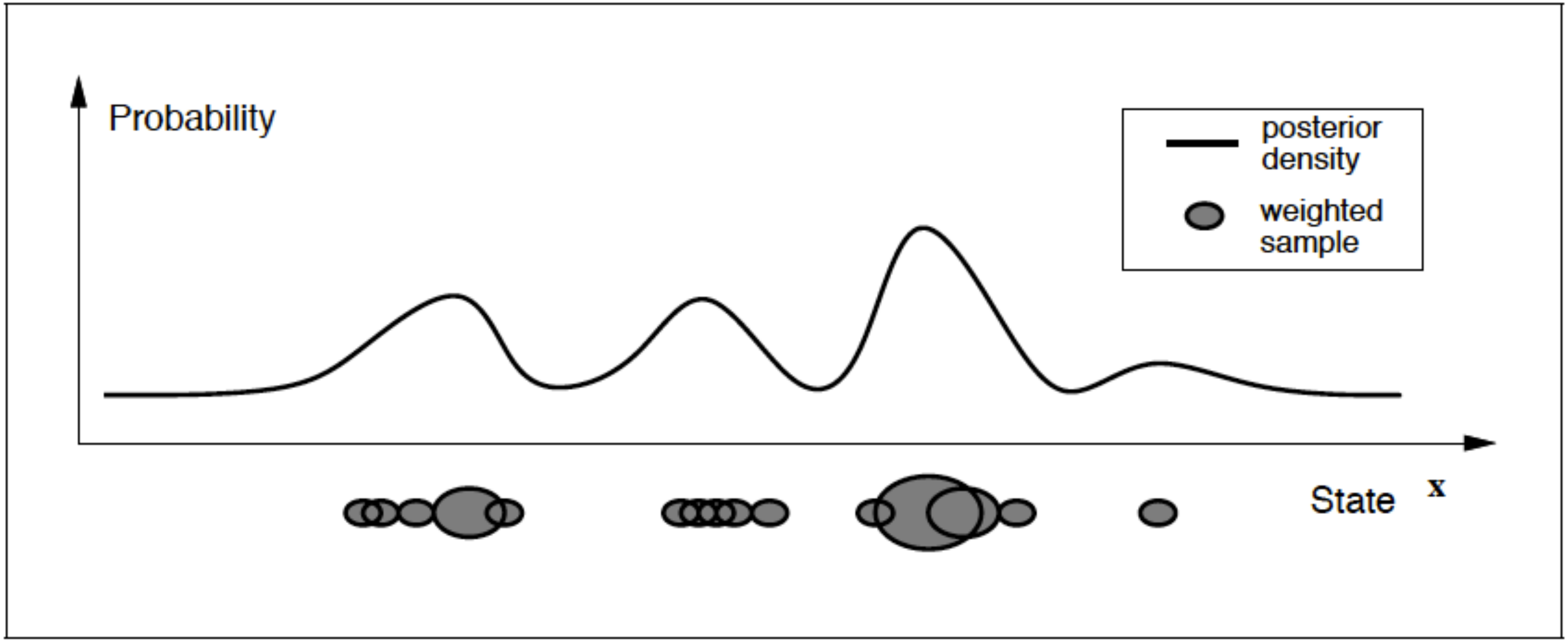
Resampling

- Next, we return to the task of generating samples from the posterior $f(x_{t+1}|y_1, \ldots, y_{t+1})$.

- It can be shown that if we resample from $\widetilde{S}_{t+1} = \{\widetilde{x}_{t+1}^1, \widetilde{x}_{t+1}^2, \ldots, \widetilde{x}_{t+1}^n\}$, with probabilities given by
$$\{\widetilde{w}_{t+1}^1, \widetilde{w}_{t+1}^2, \ldots, \widetilde{w}_{t+1}^n\},$$
then, the resulting values are approximately samples from the posterior (when n is large).

- Generating n such resamples, and calling them x_{t+1}^i, we obtain the sample set
$$S_{t+1} = \{x_{t+1}^1, x_{t+1}^2, \ldots, x_{t+1}^n\}.$$

Factor Sampling



CONDENSATION: conditional density propagation for visual tracking

International Journal of Computer Vision, 1998

Sequential Monte Carlo Algorithm

- **Algorithm 40 (Classical SMC)**

  1. Generate n samples $x_0^i \sim f(x_0)$. Set $t = 0$.

  2. **Prediction:** Generate the prediction set using:

  $$\tilde{x}_{t+1}^i \sim f(x_{t+1} \mid x_t^i), i = 1,2,\ldots,n.$$

  3. **Update:** Compute the weights $w_{t+1}^i = f(y_{t+1} \mid \tilde{x}_{t+1}^i)$,and normalize them using $\tilde{w}_{t+1}^i = w_{t+1}^i / \sum_{j=1}^n w_{t+1}^j$.

  (a) Estimate θ_{t+1} using

  $$\hat{\theta}_{t+1} = \sum_{i=1}^n g(\tilde{x}_{t+1}^i)\tilde{w}_{t+1}^i.$$

  (b) Resample from the set $\{\tilde{x}_{t+1}^1, \tilde{x}_{t+1}^2, \ldots, \tilde{x}_{t+1}^n\}$ with probabilities $\{\tilde{w}_{t+1}^1, \tilde{w}_{t+1}^2, \ldots, \tilde{w}_{t+1}^n\}$ n times to obtain the samples $x_{t+1}^i, i = 1,2,\ldots,n$.

  4. Set $t = t + 1$, and return to Step 2.

Illustration