Math508 Homework 12

Yu Huang

2007-04-26

Abstract

Continuous state space Filtering

1 Problem 3

1.1 Solve integral by Riemann sum

The difficult part is how to integrate the probability function. The idea is to replace the integral with a Riemann sum over the support where probability mass is not trivial. Based on the model, the signal X_n is concentrated around 7 with small variance. As the sequence progresses, the general trend of X_n goes upward though it's slow (due to the fact $X_{n+1} = 1.004X_n + 0.06X_nVn$).

$$\int_a^b f(x)dx = \sum_{x_i=a}^{x_i<b} f(x_i)\Delta x \tag{1}$$

The support we pick is $[-20, 30]$. Different Δx were tried. Smaller Δx, smaller mean squared error.

Figure 1 is $\Delta x = 0.15$. Figure 2 is $\Delta x = 0.34$. $\Delta x = 0.5, 0.35$ could result in zero float division in normal density.

1.2 algorithm of partical filter or sequential importance resampling

The idea is

$$E(F(\mathbf{X}_t)) = \int F_t(\mathbf{X}_t)\pi_t(\mathbf{X}_t|\mathbf{Y}_t)d\mathbf{X}_t = \int F_t(\mathbf{X}_t)\frac{\pi_t(\mathbf{X}_t|\mathbf{Y}_t)}{P(\mathbf{X}_t|\mathbf{Y}_{t-1})}P(\mathbf{X}_t|\mathbf{Y}_{t-1})d\mathbf{X}_t \tag{2}$$

$$\pi_t(\mathbf{X}_t|\mathbf{Y}_t) = \frac{P(X_t, Y_t, \mathbf{X}_{t-1}|\mathbf{Y}_{t-1})}{P(Y_t)} \tag{3}$$

$$P(X_t, Y_t, \mathbf{X}_{t-1}|\mathbf{Y}_{t-1}) = P(Y_t|X_t)P(X_t|\mathbf{X}_{t-1})P(\mathbf{X}_{t-1}|\mathbf{Y}_{t-1}) \tag{4}$$

$$P(\mathbf{X}_{t-1}|\mathbf{Y}_{t-1}) = \pi_t(\mathbf{X}_{t-1}|\mathbf{Y}_{t-1}) \tag{5}$$

$$\pi_t(\mathbf{X}_t|\mathbf{Y}_t) = \frac{P(Y_t|X_t)P(X_t|X_{t-1})\pi_t(\mathbf{X}_{t-1}|\mathbf{Y}_{t-1})}{P(Y_t)} \tag{6}$$

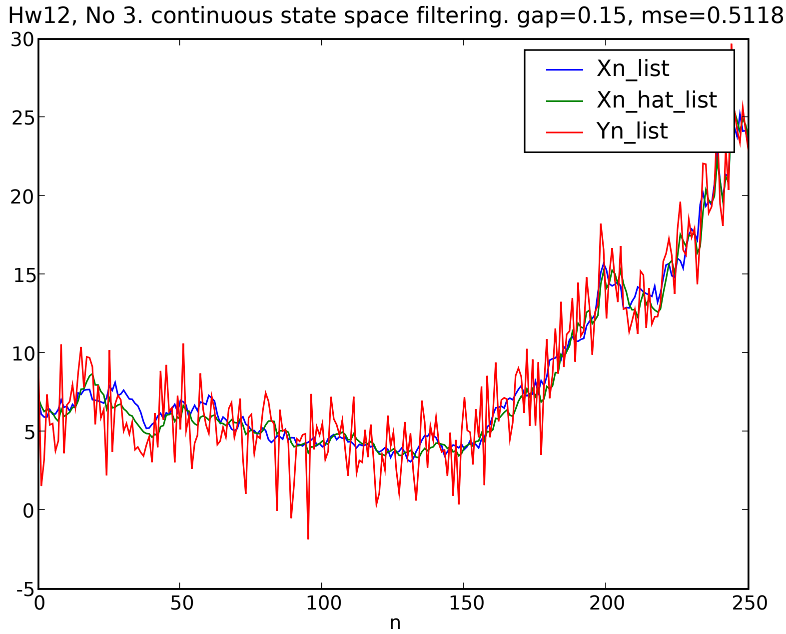

Figure 1:

The proposal distribution is $P(\mathbf{X}_t|\mathbf{Y}_{t-1}) = P(X_t|X_{t-1})\pi_t(\mathbf{X}_{t-1}|\mathbf{Y}_{t-1})$. The importance weight is $\pi_t(\mathbf{X}_t|\mathbf{Y}_t)/P(\mathbf{X}_t|\mathbf{Y}_{t-1}) = P(Y_t|X_t)/P(Y_t)$. As $P(Y_t)$ could be regarded as constant, the weight is just $P(Y_t|X_t)$.

For $t = 0$, the proposal distribution is $P(X_0)$ and the importance weight is $P(Y_0|X_0)$.

The algorithm looks like

- sample x_t^i from $P(X_t|X_{t-1})$ for $i = 1, ..., n$
- weight each sample by $w(x_t^i) = P(Y_t|x_t^i)$
- estimate is $\hat{x}_t = \sum_{i=1}^n x_t^i * w(x_t^i)$. Note: it's not the x_t^i with maximum weight cuz it simply picks the one closest to the observation, Y_t.
- resample from $\{x_t^1, ..., x_t^n\}$ with probability proportional to $w(x_t^i)$ to produce a random sample $\{x_t^{*1}, ..., x_t^{*n}\}$

Figure 3 is the result by SIR and it's pretty good (mse=0.4908) with 10000 samplings each iteration.

2 Figures

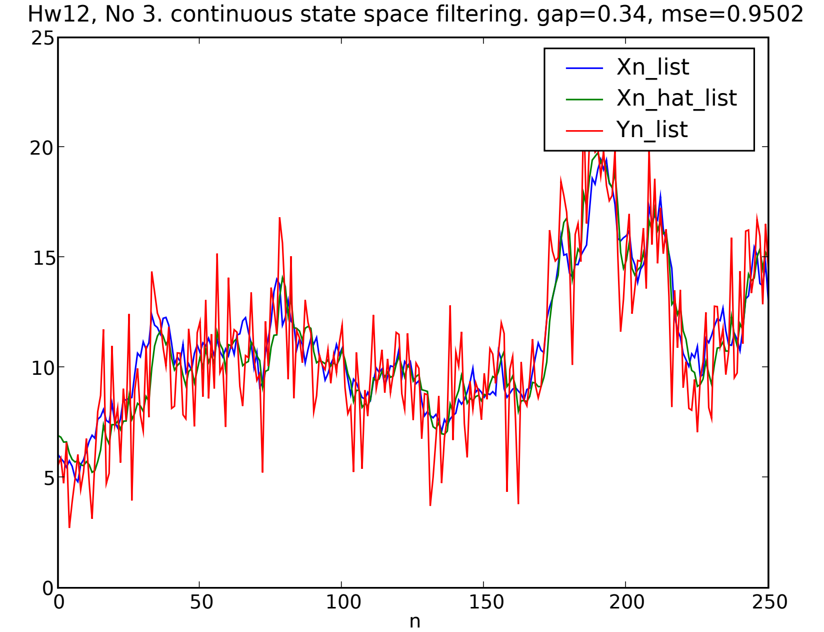

Figure 2:

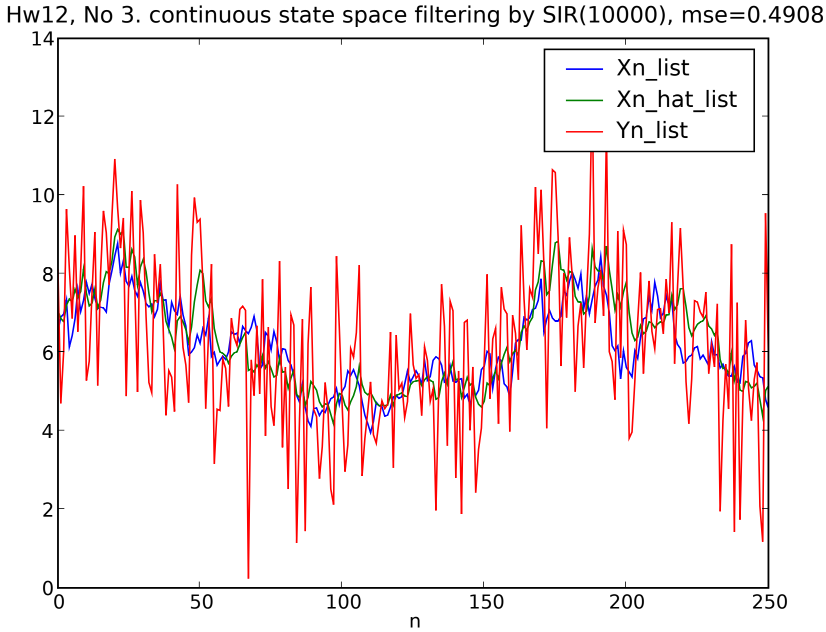

Figure 3: