Short Course

State Space Models, Generalized Dynamic Systems
and
Sequential Monte Carlo Methods,
and
their applications
in Engineering, Bioinformatics and Finance

Rong Chen

Rutgers University

Peking University

1.2 Review: Kalman Filter

Linear and Gaussian System:

$$\textbf{state equation:} \quad x_t = H_t x_{t-1} + W_t w_t \quad \textbf{where } w_t \sim N(0, I)$$
$$\textbf{observation equation:} \quad y_t = G_t x_t + V_t v_t \quad \textbf{where } v_t \sim N(0, I).$$

Examples:

- **Local level structural model**

$$\textbf{state equation} \quad m_t = m_{t-1} + \varepsilon_t$$
$$\textbf{observation equation} \quad y_t = m_t + e_t$$

  – **Example:** y_t**: realized volatility.** m_t **underlying true volatility**

- **(random) varying coefficient linear models**

$$\textbf{state equation} \quad \beta_{i,t} = \beta_{i,t-1} + \varepsilon_{i,t}$$

$$\textbf{observation equation} \quad y_t = \sum_{i=1}^{d} \beta_{i,t} x_{i,t} + e_t$$

  - **Example: varying beta in CPAM:**

$$y_t = \alpha_t + \beta_t M_t + e_t, \quad \alpha_t = \alpha_{t-1} + \varepsilon_{1,t} \quad \beta_t = \beta_{t-1} + \varepsilon_{2,t}$$

- **AR process observed with noise**

$$\textbf{state} \begin{bmatrix} x_{t-p+1} \\ x_{t-p+2} \\ \vdots \\ x_{t-1} \\ x_t \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ \phi_p & \phi_{p-1} & \phi_{p-2} & \cdots & \phi_1 \end{bmatrix} \begin{bmatrix} x_{t-p} \\ x_{t-p+1} \\ \vdots \\ z_{x-2} \\ x_{t-1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \varepsilon_t$$

$$\textbf{observation} \quad y_t = x_t + e_t$$

- **ARIMA models:** $\phi(B)x_t = \theta(B)\ \varepsilon_t$

$$x_t = \phi_1 x_{t-1} + \ldots + \phi_p x_{t-p} + \ \varepsilon_t + \theta_1\ \varepsilon_{t-1} + \ldots + \theta_q\ \varepsilon_{t-q}.$$

Let $\phi(B)z_t = \ \varepsilon_t$ **and** $x_t = \theta(B)z_t$**, then** $\phi(B)x_t = \theta(B)\ \varepsilon_t$**.**

Assume $q < p$**.**

$$\begin{bmatrix} z_{t-p+1} \\ z_{t-p+2} \\ \vdots \\ z_{t-1} \\ z_t \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ \phi_p & \phi_{p-1} & \phi_{p-2} & \cdots & \phi_1 \end{bmatrix} \begin{bmatrix} z_{t-p} \\ z_{t-p+1} \\ \vdots \\ z_{t-2} \\ z_{t-1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \varepsilon_t$$

and

$$x_t = [\theta_{p-1}, \theta_{p-2}, \ldots, \theta_1, 1] \begin{bmatrix} z_{t-p+1} \\ z_{t-p} \\ \vdots \\ z_{t-1} \\ z_t \end{bmatrix}$$

Linear and Gaussian System:

state equation: $x_t = H_t x_{t-1} + W_t w_t$ **where** $w_t \sim N(0, I)$

observation equation: $y_t = G_t x_t + V_t v_t$ **where** $v_t \sim N(0, I)$**.**

Under this model, we have

$$p(x_t \mid y_1, \ldots, y_t) \sim N(\mu_t, \Sigma_t)$$

How to obtain μ_t and Σ_t (recursively)?

Two useful facts about joint Normal distribution

(1) If $(X, Y) \sim N((\mu_x, \mu_y), \Sigma)$**, then**

$$
\begin{aligned}
E(X \mid Y) &= \mu_x - \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y) \\
V(X \mid Y) &= \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}
\end{aligned}
$$

(2) If $X \sim N(\mu_x, \Sigma_x)$ **and** $Y = GX + Vv$ **where** $v \sim N(0, I)$**, what is** $p(X \mid Y) \propto p(Y \mid X)p(X)$**?**

First, find the joint distribution of $(X, Y) \sim N((\mu_x, \mu_y), \Sigma)$

$$
\begin{aligned}
\mu_x &= \mu_x \quad \textbf{and} \qquad \Sigma_{xx} = \Sigma_x \\
\mu_y &= E[Y] = E[GX + Vv] = GE[X] = G\mu_x \\
\Sigma_{xy} &= E[(X - \mu_x)(Y - \mu_y)^{'}] = E[(X - \mu_x)((X - \mu_x)^{'}G^{'} + v^{'}V^{'})] = \Sigma_x G^{'} \\
\Sigma_{yx} &= G\Sigma_x \\
\Sigma_{yy} &= E[(Y - \mu_y)(Y - \mu_y)^{'}] = E[G(X - \mu_x)(X - \mu_x)^{'}G^{'} + Vvv^{'}V^{'}] \\
&= G\Sigma_x G^{'} + VV^{'}
\end{aligned}
$$

Kalman Filter:

Suppose at time $t-1$ **we have obtained** μ_{t-1} **and** Σ_{t-1}**. That is,**

$$p(x_{t-1} \mid y_1, \ldots, y_{t-1}) \sim N(\mu_{t-1}, \Sigma_{t-1}).$$

- **Before we observe** y_t**, we can use the state equation to predict** x_t**. That is,**

  $$p(x_t \mid y_1, \ldots, y_{t-1}) \sim N(\mu_t^{t-1}, \Sigma_t^{t-1})$$

  Note:

  $$\begin{aligned} p(x_t \mid y_1, \ldots, y_{t-1}) &= \int p(x_t \mid x_{t-1}, y_1, \ldots, y_{t-1}) dx_{t-1} \\ &= \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid y_1, \ldots, y_{t-1}) dx_{t-1} \end{aligned}$$

  Since

  $$x_t = H_t x_{t-1} + W_t w_t, \quad \textbf{we have} \quad x_t \sim N(H_t \mu_{t-1}, H_t \Sigma_{t-1} H_t^{'} + W_t W_t^{'})$$

  Hence $\mu_t^{t-1} = H_t \mu_{t-1}$**,** $\Sigma_t^{t-1} = H_t \Sigma_{t-1} H_t^{'} + W_t W_t^{'}$

- **The observation equation says:**

$$y_t = G_t x_t + V_t v_t$$

It provides additional information about x_t **—- or correction to the prediction.**

- **Bayes Theorem**

$$p(X \mid Y) \propto p(Y \mid X) p(X)$$

or

$$\begin{aligned} p(x_t \mid y_1, \ldots, y_t) &\propto p(y_t \mid x_t, y_1, \ldots, y_{t-1}) p(x_t \mid y_1, \ldots, y_{t-1}) \\ &= p(y_t \mid x_t) p(x_t \mid y_1, \ldots, y_{t-1}) \end{aligned}$$

We have

$$\mu_t = \mu_t^{t-1} + K_t (y_t - G_t \mu_t^{t-1}) \qquad \Sigma_t = \Sigma_t^{t-1} - K_t G_t \Sigma_t^{t-1}$$

where $K_t = \Sigma_t^{t-1} G_t' [G_t' \Sigma_t^{t-1} G_t + V'V]^{-1}$**.**

Summary:

state equation: $x_t = H_t x_{t-1} + W_t w_t$ **where** $w_t \sim N(0, I)$

observation equation: $y_t = G_t x_t + V_t v_t$ **where** $v_t \sim N(0, I)$**.**

Kalman Filter: $(\mu_{t-1}, \Sigma_{t-1})$ **to** (μ_t, Σ_t)

$$
\begin{aligned}
\mu_t^{t-1} &= H_t \mu_{t-1} \\
\Sigma_t^{t-1} &= H_t \Sigma_{t-1} H_t^{'} + W_t W_t^{'} \\
\mu_t &= \mu_t^{t-1} + K_t (y_t - G_t \mu_t^{t-1}) \\
\Sigma_t &= \Sigma_t^{t-1} - K_t G_t \Sigma_t^{t-1}
\end{aligned}
$$

where

$$K_t = \Sigma_t^{t-1} G_t^{'} [G_t^{'} \Sigma_t^{t-1} G_t + V^{'} V]^{-1}.$$

Note:

- **Kalman filter can be used to calculate the likelihood function**
- **Hence often used as an estimation tool**
- **It can also do smoothing** $p(x_t \mid y_1, \ldots, y_n)$ **and prediction** $p(x_{t+d} \mid y_1, \ldots, y_t)$**.**

For nonlinear systems: Use approximation

- **Extended Kalman filter**
- **etc**

Kalman Filter R implementation:

- *KalmanLike*
- *KalmanRun*
- *KalmanSmooth*
- *KalmanForecast*

1.3 Review: Basic Monte Carlo Methods

Statistical Inferences:

- **Distribution:** $p(\cdot, \theta)$ **with unknown** θ
- **Objective: estimate** θ**.**
- **Observe:** $X_1, \ldots, X_n$ **i.i.d.**
- **Method: M.L.E. or other**

Monte Carlo Methods:

- **Distribution:** $p(\cdot, \theta)$ **with known** θ
- **Objective: calculate** $E(h(X))$
- **Generate:** $X_1, \ldots, X_n$ **i.i.d from** $p(\cdot, \theta)$
- **Method:** $\sum_{i=1}^{n} h(X_i)/n$ **(or improved version)**

Simple methods of generating random samples:

(1) Transformation:

If $Y = f(X)$**, and** $x \sim X$**, then** $y = f(x) \sim Y$

- **Example: Normal(0,1):** $Y = \sqrt{-2ln(X_1)}cos(2X_2)$ **where** X_1**,** X_2 **independent Uniform(0,1)**
- **Example: Normal(**μ, σ^2**).** $Y = \mu + \sigma X$**, where** $X \sim N(0,1)$
- **Example:** χ^2_k**.** $Y = \sum_{i=1}^{k} X_i^2$**, where** $X_i \sim N(0,1)$**, independent.**

(2) Inverse CDF:

If X **has a cdf** F**, then** $F(X) \sim U(0,1)$**.**

If $F(x)$ **is strictly increase (in the range)**

then $F^{-1}(U) \sim X$ **where** $U \sim U(0,1)$**.**

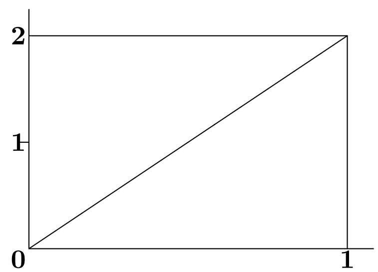
Example: pdf: $p(x) = 2x$**, (**$0 < x < 1$**).**

CDF: $F(x) = x^2, 0 < x < 1$

Hence $X = \sqrt{U}$**,** $U \sim U(0,1)$**.**

(3) Rejection method:

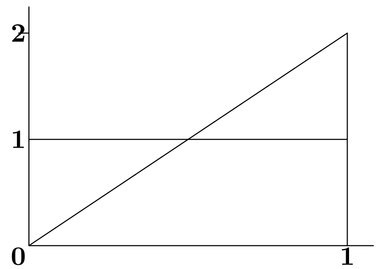
Example: pdf: $p(x) = 2x$, $(0 < x < 1)$.



- **Sample uniform points in the area.**
- **Accept the points under the density curve.**
- **The x-coordinate of the accepted points** $\sim X$.

(4) Importance Sampling:

Example: pdf: $p(x) = 2x$, $(0 < x < 1)$.



- **In the over-presented area, down weight the sample.**
- **In the under-presented area, up weight the sample**

How?

Target distribution π; a sample $x_1, \ldots, x_m$ from g.

$$E_\pi(f(X)) = \int f(x)\pi(x)dx = \int f(x)\frac{\pi(x)}{g(x)}g(x)dx = E_g(f(X)w(X))$$

where $w(x) = \pi(x)/g(x)$.

We have

$$\frac{1}{m}\sum_{i=1}^{m} w(x_i)f(x_i) \approx E_\pi(f(x))$$

Let weight $w_i \propto \pi(x_i)/g(x_i)$, we can use

$$\frac{1}{\sum w_i}\sum_{i=1}^{m} w_i f(x_i) \approx E_\pi(f(x))$$

<u>**Efficiency:**</u>

$$\textbf{effective sample size} = \frac{m}{1 + cv^2(w)}$$

Example: $m = 100$. **use** $U(0,1)$**: ESS**$= 78$**;** $N(0,1)$**: ESS**$= 24$

(5) **<u>Sequential Sampling</u>** $(X, Y) \sim p(x, y)$**.**

(i) : Sample $X = x$ **from the marginal distribution** $p(x) = \int p(x, y) dy$

(ii) : Sample $Y = y$ **from the conditional distribution** $p(y \mid X = x) = p(x, y)/p(x)$

Example:

$$(X, Y) \sim N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}\right)$$

(i) $X = x$ **from** $N(\mu_1, \sigma_1^2)$

(ii) $Y = y$ **from** $N(\mu_2 - \rho\frac{\sigma_2}{\sigma_1}(x - \mu_1), (1 - \rho)\sigma_2^2)$**.**

Example: Time Series $X_t = \phi X_{t-1} + e_t$ **where** $e_t \sim N(0, \sigma^2)$

(1) X_0 **from** $N(\mu_0, \sigma_0^2)$ **(often stationary dist)**

(2) X_t **from** $N(\phi X_{t-1}, \sigma^2)$

(Augmentation:) Use sequential sampling when:
$p_Y(y)$ **is not easy, but** $p_X(x)$ **and** $p(Y \mid X = x)$ **are easy,**

Example: $Y = X_1 + \ldots + X_N$,
where X_i $i.i.d. \sim Bernolli(p)$**, and** $N \sim Poisson(\lambda)$**.**
(i) Sample $N = n \sim Poisson(\lambda)$
(ii) Sample Y **from Binomial**(n, p)

Example: $Y \sim pN(\mu_0, \sigma_0^2) + (1 - p)N(\mu_1, \sigma_1^2)$
(i) Sample $I = i$ **from Bernoulli(p)**
(ii) Sample Y **from** $N(\mu_i, \sigma_i^2)$

(6) Gibbs Sampler:

- $p(X, Y)$ **difficult, but** $p(X \mid Y)$ **and** $p(Y \mid X)$ **easy**
- **Initial values:** $X = x^{(0)}, Y = y^{(0)}$**.**

  Iteratively for $i = 1 \ldots$**, do**

  (i) Sample $X = x^{(i+1)}$ **from** $p(X \mid Y = y^{(i)})$

  (ii) Sample $Y = y^{(i+1)}$ **from** $p(Y \mid X = x^{(i+1)})$
- **After a burn-out period:** $i = 0, \ldots, m$**, the samples** $(x^{(i)}, y^{(i)})$**,** $i = m + 1, \ldots,$ **are** *correlated* **samples from** $p(X, Y)$**.**

Other more advanced Markov Chain Monte Carlo methods