

Конспект лекции

«Линейные динамические системы. Фильтр Калмана.»

по спецкурсу «Структурные методы анализа изображений и сигналов» 2011

Ликбез: некоторые свойства нормального распределения.

Пусть $\mathbf{x} \in \mathbb{R}^d$ распределен по нормальному закону, т.е.

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{2\pi}^d \sqrt{\det \Sigma}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right).$$

Разобьем вектор \mathbf{x} на две группы переменных $\mathbf{x}_a, \mathbf{x}_b$ и обозначим

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{bmatrix}, \quad \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix}, \quad \Lambda = \Sigma^{-1} = \begin{bmatrix} \Lambda_{aa} & \Lambda_{ab} \\ \Lambda_{ba} & \Lambda_{bb} \end{bmatrix}.$$

Тогда можно показать, что

$$\begin{aligned} p(\mathbf{x}_a) &= \mathcal{N}(\mathbf{x}_a|\boldsymbol{\mu}_a, \Sigma_{aa}), \\ p(\mathbf{x}_a|\mathbf{x}_b) &= \mathcal{N}(\mathbf{x}_a|\boldsymbol{\mu}_a - \Lambda_{aa}^{-1}\Lambda_{ab}(\mathbf{x}_b - \boldsymbol{\mu}_b), \Lambda_{aa}^{-1}). \end{aligned} \tag{1}$$

Это, в частности, означает, что у многомерного нормального распределения все маргинальные и маргинальные условные распределения также являются нормальными.

Рассмотрим величину $\mathbf{y} \in \mathbb{R}^D$, которая с точностью до нормального шума связана линейно с величиной \mathbf{x} , т.е.

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|A\mathbf{x}, \Gamma).$$

Тогда можно показать, что

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|A\boldsymbol{\mu}, \Gamma + A\Sigma A^T), \tag{2}$$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}|P(A^T\Gamma^{-1}\mathbf{y} + \Sigma^{-1}\boldsymbol{\mu}), P), \quad P = (\Sigma^{-1} + A^T\Gamma^{-1}A)^{-1}. \tag{3}$$

В частности, если $\Gamma = 0$, то результат (2) говорит о том, что любые линейные комбинации компонент нормального распределения также распределены нормально.

1. Модельный пример задачи фильтрации сигнала.

Рассмотрим модельную задачу сопровождения (трекинга) объекта. Пусть имеется некоторая траектория объекта в пространстве (см. рис. 1а). При этом координаты объекта в каждый момент времени измеряются с некоторой погрешностью (см. рис. 1б, красная кривая). Задача состоит в том, чтобы уточнить координаты объекта путем сглаживания наблюдаемой траектории (см. рис. 1б, зеленая кривая).

Обозначим через $\mathbf{x}_1, \dots, \mathbf{x}_N$ наблюдаемые характеристики объекта в моменты времени $1, \dots, N$, а через $\mathbf{t}_1, \dots, \mathbf{t}_N$ — скрытые (истинные) параметры объекта. Предположим, что динамика изменения параметров объекта во времени является марковским процессом, т.е. величина \mathbf{t}_n зависит только от \mathbf{t}_{n-1} , а наблюдаемые характеристики \mathbf{x}_n полностью определяются параметрами объекта \mathbf{t}_n в момент времени n . Таким образом, мы получили байесовскую сеть, показанную на рис. 2, где $p(\mathbf{t}_n|\mathbf{t}_{n-1})$ — модель движения объекта, а $p(\mathbf{x}_n|\mathbf{t}_n)$ — модель сенсора.

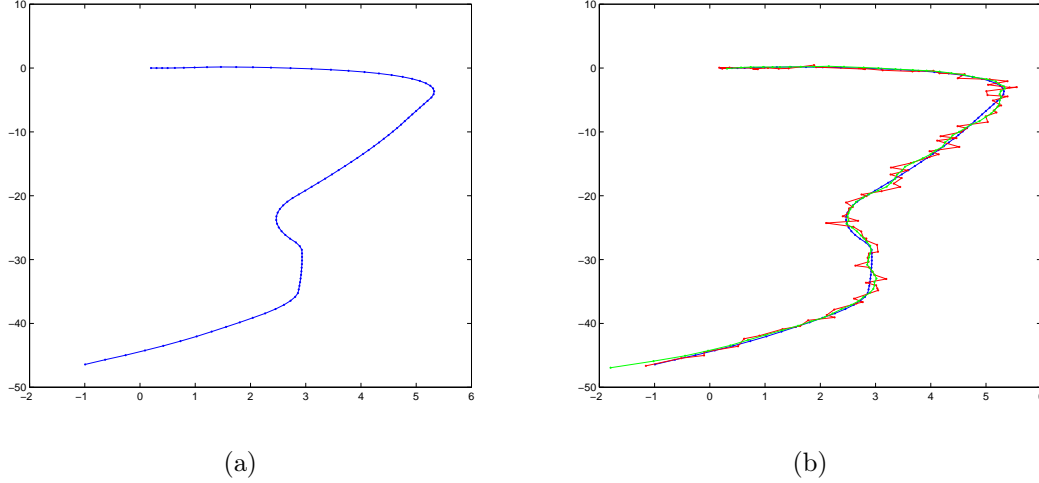


Рис. 1: Траектория движения некоторого объекта на плоскости. Синяя кривая показывает истинную траекторию объекта, красная кривая — наблюдаемая траектория, зеленая кривая — сглаженная траектория.

Рассмотрим в качестве параметров объекта координаты, скорости и ускорения по каждой координате $\mathbf{t}_n = [\xi_1(n), \dot{\xi}_1(n), \ddot{\xi}_1(n), \xi_2(n), \dot{\xi}_2(n), \ddot{\xi}_2(n)]$. Тогда моделировать движение объекта можно следующим образом:

$$\begin{aligned}\xi_i(n) &= \xi_i(n-1) + \dot{\xi}_i(n-1)\Delta t + \ddot{\xi}_i(n-1)\frac{\Delta t^2}{2} + \varepsilon_{1i}, \quad i = 1, 2; \\ \dot{\xi}_i(n) &= \dot{\xi}_i(n-1) + \ddot{\xi}_i(n-1)\Delta t + \varepsilon_{2i}, \quad i = 1, 2; \\ \ddot{\xi}_i(n) &= \ddot{\xi}_i(n-1) + \varepsilon_{3i}, \quad i = 1, 2; \\ \varepsilon_{ji} &\sim \mathcal{N}(0, \gamma_{ji}), \quad j = 1, 2, 3, \quad i = 1, 2.\end{aligned}$$

Аналогично, модель сенсора можно представить как

$$\mathbf{x}_i(n) = \xi_i(n) + \nu_i, \quad \nu_i \sim \mathcal{N}(0, \sigma_i), \quad i = 1, 2.$$

Не ограничивая общности, можно считать, что $\Delta t = 1$. Тогда модель движения и модель сенсора можно записать в матричном виде следующим образом:

$$\begin{aligned}\mathbf{t}_n &= A\mathbf{t}_{n-1} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \Gamma), \quad \Leftrightarrow p(\mathbf{t}_n|\mathbf{t}_{n-1}) = \mathcal{N}(A\mathbf{t}_{n-1}, \Gamma), \\ \mathbf{x}_n &= C\mathbf{t}_n + \boldsymbol{\nu}, \quad \boldsymbol{\nu} \sim \mathcal{N}(0, \Sigma), \quad \Leftrightarrow p(\mathbf{x}_n|\mathbf{t}_n) = \mathcal{N}(C\mathbf{t}_n, \Sigma), \\ A &= \begin{pmatrix} 1 & 1 & 0.5 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0.5 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}, \\ \Gamma &= \text{diag}(\gamma_{11}, \dots, \gamma_{32}), \quad \Sigma = \text{diag}(\sigma_1, \sigma_2).\end{aligned}$$

2. Линейная динамическая система.

Линейной динамической системой (ЛДС) называется байесовская сеть, показанная на рис. 2, где $\mathbf{x}_n \in \mathbb{R}^d$, $\mathbf{t}_n \in \mathbb{R}^D$, и все атомарные распределения задаются линейной гауссовской моделью:

$$\begin{aligned}p(\mathbf{t}_n|\mathbf{t}_{n-1}) &= \mathcal{N}(A\mathbf{t}_{n-1}, \Gamma), \\ p(\mathbf{x}_n|\mathbf{t}_n) &= \mathcal{N}(C\mathbf{t}_n, \Sigma), \\ p(\mathbf{t}_1) &= \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0).\end{aligned}\tag{4}$$

Заметим, что байесовская сеть на рис. 2 соответствует также скрытой марковской модели. Основное отличие ЛДС от СММ заключается в том, что в ЛДС переменные \mathbf{t}_n являются непрерывными, а в СММ — дискретными. Совместное распределение всех переменных в ЛДС задается как

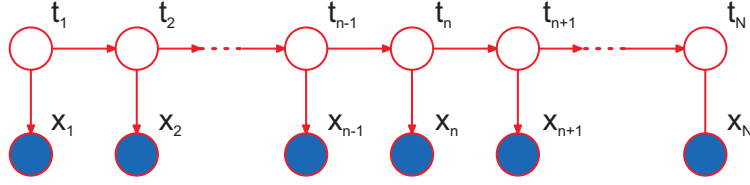


Рис. 2: Графическая модель линейной динамической системы.

$$p(X, T|A, \Gamma, C, \Sigma) = p(t_1)p(x_1|t_1) \prod_{n=2}^N p(t_n|t_{n-1})p(x_n|t_n) \propto \exp \left[-\frac{1}{2} \left(\sum_{n=1}^N (x_n - Ct_n)^T \Sigma^{-1} (x_n - Ct_n) + \sum_{n=2}^N (t_n - At_{n-1})^T \Gamma^{-1} (t_n - At_{n-1}) + (t_1 - \mu_0)^T \Sigma_0^{-1} (t_1 - \mu_0) \right) \right].$$

В показателе экспоненты стоит квадратичная функция относительно переменных модели. Следовательно, совместное распределение $p(X, T)$ является многомерным нормальным распределением. Тогда из свойства (1) следует, что распределение $p(T|X)$, а также все маргинальные распределения вида $p(t_n|X)$, $p(t_n|x_1, \dots, x_n)$ также являются нормальными.

У нормального распределения математическое ожидание совпадает с модой. Это означает, что в ЛДС наиболее вероятная конфигурация T при известном X определяется математическим ожиданием нормального распределения $p(T|X)$. Рассмотрим маргинальное распределение $p(t_n|X)$. Из свойства (1) следует, что математическое ожидание $p(t_n|X)$ определяется соответствующей компонентой математического ожидания распределения $p(T|X)$. Таким образом, в линейной динамической системе знание маргинальных распределений $p(t_n|X)$ позволяет найти и наиболее вероятную конфигурацию всех скрытых переменных модели T . В результате, для ЛДС аналог алгоритма Витерби не требуется. Заметим, что в СММ, в отличие от ЛДС, наиболее вероятная конфигурация T не состоит, вообще говоря, из индивидуально наиболее вероятных состояний $t_n^* = \arg \max p(t_n|X)$.

3. Вывод в ЛДС: фильтр Калмана.

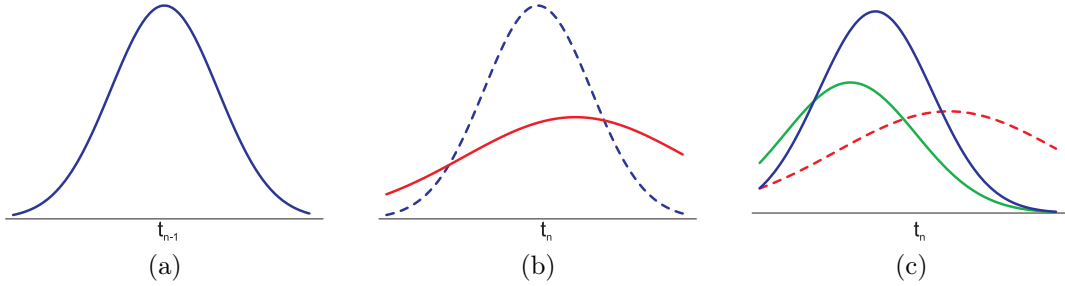


Рис. 3: Прогнозирование с помощью фильтра Калмана. На рис. а показано текущее распределение для t_{n-1} (синяя кривая), на рис. б показано прогнозное распределение для t_n (красная кривая), на рис. в показан уточненный прогноз для t_n после прихода значения x_n .

Рассмотрим задачу фильтрации сигнала в реальном времени. Это соответствует поиску распределений $p(t_n|x_1, \dots, x_n)$ для каждого момента времени $n = 1, \dots, N$. Как было показано выше, все эти распределения являются нормальными:

$$p(t_n|x_1, \dots, x_n) = \mathcal{N}(t_n|\mu_n, V_n).$$

Пусть известно распределение $p(t_{n-1}|x_1, \dots, x_{n-1})$ для момента времени $n - 1$. Тогда прогноз значения t_n вычисляется следующим образом:

$$p(t_n|x_1, \dots, x_{n-1}) = \int p(t_n, t_{n-1}|x_1, \dots, x_{n-1}) dt_{n-1} = \int p(t_n|t_{n-1})p(t_{n-1}|x_1, \dots, x_{n-1}) dt_{n-1} = \int \mathcal{N}(t_n|At_{n-1}, \Gamma) \mathcal{N}(t_{n-1}|\mu_{n-1}, V_{n-1}) dt_{n-1} = \mathcal{N}(t_n|A\mu_{n-1}, \Gamma + AV_{n-1}A^T).$$

Последнее равенство следует из свойства (2) для нормальных распределений. Таким образом,

$$p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1}) = \mathcal{N}(\mathbf{t}_n | \tilde{\boldsymbol{\mu}}_n, \tilde{V}_n), \quad \tilde{\boldsymbol{\mu}}_n = A\boldsymbol{\mu}_{n-1}, \quad \tilde{V}_n = \Gamma + AV_{n-1}A^T. \quad (5)$$

После того, как значение \mathbf{x}_n становится известным, можно уточнить прогноз для \mathbf{t}_n :

$$\begin{aligned} p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_n) &= \frac{p(\mathbf{t}_n, \mathbf{x}_1, \dots, \mathbf{x}_n)}{p(\mathbf{x}_1, \dots, \mathbf{x}_n)} = \frac{p(\mathbf{x}_n | \mathbf{t}_n) p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1} | \mathbf{t}_n) p(\mathbf{t}_n)}{p(\mathbf{x}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1}) p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1})} = \\ &= \frac{p(\mathbf{x}_n | \mathbf{t}_n) p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1})}{p(\mathbf{x}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1})} \propto \mathcal{N}(\mathbf{x}_n | C\mathbf{t}_n, \Sigma) \mathcal{N}(\mathbf{t}_n | \tilde{\boldsymbol{\mu}}_n, \tilde{V}_n) = \mathcal{N}(\mathbf{t}_n | \boldsymbol{\mu}_n, V_n). \\ \boldsymbol{\mu}_n &= \tilde{\boldsymbol{\mu}}_n + K_n(\mathbf{x}_n - C\tilde{\boldsymbol{\mu}}_n), \\ V_n &= (I - K_n C) \tilde{V}_n, \\ K_n &= \tilde{V}_n C^T (C \tilde{V}_n C^T + \Sigma)^{-1}. \end{aligned} \quad (6)$$

Этот результат следует из свойства (3) для нормальных распределений.

Таким образом, фильтр Калмана состоит из двух шагов. Пусть имеется текущее (априорное) распределение $p(\mathbf{t}_{n-1} | \mathbf{x}_1, \dots, \mathbf{x}_{n-1})$ (см. рис. 3а). На первом шаге осуществляется прогноз значения \mathbf{t}_n по формулам (5) (см. рис. 3б). При этом дисперсия прогноза (матрица \tilde{V}_n) увеличивается по сравнению с дисперсией для \mathbf{t}_{n-1} . Затем, на втором шаге, происходит коррекция прогноза для \mathbf{t}_n с учетом новой информации \mathbf{x}_n (формулы (6)). При этом дисперсия прогноза V_n уменьшается по сравнению с \tilde{V}_n (см. рис. 3с).

4. Вывод в ЛДС: РТС уравнения.

Рассмотрим задачу фильтрации в случае, когда наблюдаемый сигнал X известен полностью до начала процедуры фильтрации. Эта задача соответствует поиску распределений $p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_N)$. Такие распределения также нужны для решения задачи обучения параметров ЛДС ($A, C, \Gamma, \Sigma, \boldsymbol{\mu}_0, V_0$).

Как уже было замечено выше, графическая модель ЛДС совпадает с аналогичной для СММ. Следовательно, алгоритм вывода в СММ «вперед-назад» (реализация общего алгоритма SUM-PRODUCT для графической модели типа цепочка) может быть использован и для ЛДС с тем отличием, что суммы по \mathbf{t}_n заменяются на интегралы:

$$\begin{aligned} \gamma(\mathbf{t}_n) &= p(\mathbf{t}_n | X) = \hat{\alpha}(\mathbf{t}_n) \hat{\beta}(\mathbf{t}_n) = \mathcal{N}(\mathbf{t}_n | \hat{\boldsymbol{\mu}}_n, \hat{V}_n), \\ c_n \hat{\alpha}(\mathbf{t}_n) &= p(\mathbf{x}_n | \mathbf{t}_n) \int \hat{\alpha}(\mathbf{t}_{n-1}) p(\mathbf{t}_n | \mathbf{t}_{n-1}) d\mathbf{t}_{n-1}, \end{aligned} \quad (7)$$

$$c_{n+1} \hat{\beta}(\mathbf{t}_n) = \int \hat{\beta}(\mathbf{t}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{t}_{n+1}) p(\mathbf{t}_{n+1} | \mathbf{t}_n) d\mathbf{t}_{n+1}. \quad (8)$$

Проход вперед в этом алгоритме, т.е. вычисление c_n и $\hat{\alpha}(\mathbf{t}_n) = p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_n)$, в точности соответствует фильтру Калмана, рассмотренному в предыдущем пункте. Проход назад в случае ЛДС получил название РТС уравнений (по первым буквам фамилий авторов — Rauch, Tung, Striebel). Можно показать, что для прохода назад справедливы следующие формулы:

$$\begin{aligned} p(\mathbf{t}_n | X) &= \mathcal{N}(\mathbf{t}_n | \hat{\boldsymbol{\mu}}_n, \hat{V}_n), \\ \hat{\boldsymbol{\mu}}_n &= \boldsymbol{\mu}_n + J_n(\hat{\boldsymbol{\mu}}_{n+1} - A\boldsymbol{\mu}_n), \\ \hat{V}_n &= V_n + J_n(\hat{V}_{n+1} - \tilde{V}_n) J_n^T. \end{aligned}$$

По аналогии с алгоритмом «вперед-назад» для СММ, мы также можем эффективно вычислить условное распределение для соседних скрытых переменных $\mathbf{t}_{n-1}, \mathbf{t}_n$:

$$\xi(\mathbf{t}_{n-1}, \mathbf{t}_n) = p(\mathbf{t}_n, \mathbf{t}_{n-1} | X) = \frac{1}{c_n} \hat{\alpha}(\mathbf{t}_n) p(\mathbf{x}_n | \mathbf{t}_n) p(\mathbf{t}_n | \mathbf{t}_{n-1}) \hat{\beta}(\mathbf{t}_n) = \mathcal{N}([\mathbf{t}_n, \mathbf{t}_{n-1}] | [\gamma(\mathbf{t}_n), \gamma(\mathbf{t}_{n-1})], J_{n-1} \hat{V}_n).$$

Объединяя все вышесказанное, получаем общую схему алгоритма «вперед-назад» для ЛДС (см. Алгоритм 1).

5. Обучение фильтра Калмана.

Рассмотрим задачу обучения параметров ЛДС по данным. Будем решать эту задачу с помощью метода максимального правдоподобия, т.е.

$$\log p(X | A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0) = \log \int p(X, T | A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0) dT \rightarrow \max_{A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0}$$

Алгоритм 1: Алгоритм «вперед-назад» для ЛДС

Вход: $\mathbf{x}_1, \dots, \mathbf{x}_N$ – наблюдаемый сигнал, $(A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0)$ – параметры ЛДС

Выход: $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_N, V_1, \dots, V_N$ – параметры распределений $p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_n)$;

$\hat{\boldsymbol{\mu}}_1, \dots, \hat{\boldsymbol{\mu}}_N, \hat{V}_1, \dots, \hat{V}_N$ – параметры распределений $p(\mathbf{t}_n | \mathbf{x}_1, \dots, \mathbf{x}_N)$;

$L = \log p(X|A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0)$ – логарифм неполного правдоподобия.

// Проход вперед (фильтр Калмана)

$$K_1 = V_0 C^T (C V_0 C^T + \Sigma)^{-1};$$

$$c_1 = \mathcal{N}(\mathbf{x}_1 | C \boldsymbol{\mu}_0, C V_0 C^T + \Sigma);$$

$$\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + K_1 (\mathbf{x}_1 - C \boldsymbol{\mu}_0);$$

$$V_1 = (I - K_1 C) V_0;$$

для $n = 2, \dots, N$

$$\hat{V}_{n-1} = A V_{n-1} A^T + \Gamma;$$

$$K_n = \hat{V}_{n-1} C^T (C \hat{V}_{n-1} C^T + \Sigma)^{-1};$$

$$c_n = \mathcal{N}(\mathbf{x}_n | C A \boldsymbol{\mu}_{n-1}, C \hat{V}_{n-1} C^T + \Sigma);$$

$$\boldsymbol{\mu}_n = A \boldsymbol{\mu}_{n-1} + K_n (\mathbf{x}_n - C A \boldsymbol{\mu}_{n-1});$$

$$V_n = (I - K_n C) \hat{V}_{n-1};$$

// Проход назад (РТС уравнения)

$$\hat{\boldsymbol{\mu}}_N = \boldsymbol{\mu}_N;$$

$$\hat{V}_N = V_N;$$

для $n = N - 1, \dots, 1$

$$J_n = V_n A^T \hat{V}_{n+1}^{-1};$$

$$\hat{\boldsymbol{\mu}}_n = \boldsymbol{\mu}_n + J_n (\hat{\boldsymbol{\mu}}_{n+1} - A \boldsymbol{\mu}_n);$$

$$\hat{V}_n = V_n + J_n (\hat{V}_{n+1} - \hat{V}_n) J_n^T;$$

$$L = \sum_{n=1}^N \log c_n; \quad // \text{ Вычисляем логарифм неполного правдоподобия}$$

Это задача оптимизации неполного правдоподобия. Следовательно, здесь можно применить EM-алгоритм. Итерационная схема EM-алгоритма состоит из двух шагов:

E-шаг:

$$p(T|X, \Theta_{old}) = \frac{p(X, T | \Theta_{old})}{p(X | \Theta_{old})},$$

M-шаг:

$$\mathbb{E}_{T|X, \Theta_{old}} \log p(X, T | \Theta) \rightarrow \max_{\Theta}$$

Рассмотрим обучение параметров C, Σ . Запишем слагаемые $\mathbb{E} \log p(X, T | \Theta)$, которые зависят от C, Σ :

$$\begin{aligned} & -\frac{1}{2} \mathbb{E} \left[\sum_{n=1}^N (\mathbf{x}_n - C \mathbf{t}_n)^T \Sigma^{-1} (\mathbf{x}_n - C \mathbf{t}_n) \right] - \frac{N}{2} \log \det \Sigma = \\ & = -\frac{1}{2} \sum_{n=1}^N [\mathbf{x}_n^T \Sigma^{-1} \mathbf{x}_n - 2 \mathbb{E} \mathbf{t}_n^T C^T \Sigma^{-1} \mathbf{x}_n + \text{tr}(C^T \Sigma^{-1} C \mathbb{E}(\mathbf{t}_n \mathbf{t}_n^T))] - \frac{N}{2} \log \det \Sigma \rightarrow \max_{C, \Sigma} \end{aligned}$$

Здесь $\mathbb{E} \mathbf{t}_n = \hat{\boldsymbol{\mu}}_n$, $\mathbb{E} \mathbf{t}_n \mathbf{t}_n^T = \hat{V}_n + \hat{\boldsymbol{\mu}}_n \hat{\boldsymbol{\mu}}_n^T$. Вычисляя производные по C, Σ и приравнивая их к нулю, получаем следующие формулы пересчета:

$$\begin{aligned} C^{new} &= \left(\sum_{n=1}^N \mathbf{x}_n \mathbb{E} \mathbf{t}_n^T \right) \left(\sum_{n=1}^N \mathbb{E} \mathbf{t}_n \mathbf{t}_n^T \right)^{-1}, \\ \Sigma^{new} &= \frac{1}{N} \sum_{n=1}^N [\mathbf{x}_n \mathbf{x}_n^T - C^{new} \mathbb{E} \mathbf{t}_n \mathbf{x}_n^T - \mathbf{x}_n \mathbb{E} \mathbf{t}_n^T (C^{new})^T + C^{new} \mathbb{E} \mathbf{t}_n \mathbf{t}_n^T (C^{new})^T]. \end{aligned}$$

Рассуждая аналогично для остальных параметров ЛДС, получаем следующие формулы пересчета:

$$\begin{aligned}\boldsymbol{\mu}_0^{new} &= \mathbb{E}\mathbf{t}_1, \\ V_0^{new} &= \mathbb{E}\mathbf{t}_1\mathbf{t}_1^T + (\boldsymbol{\mu}_0 - \mathbb{E}\mathbf{t}_1)^T\boldsymbol{\mu}_0 - \boldsymbol{\mu}_0^T\mathbb{E}\mathbf{t}_1, \\ A^{new} &= \left(\sum_{n=2}^N \mathbb{E}\mathbf{t}_n\mathbf{t}_{n-1}^T \right) \left(\sum_{n=2}^N \mathbb{E}\mathbf{t}_{n-1}\mathbf{t}_{n-1}^T \right)^{-1}, \\ \Gamma^{new} &= \frac{1}{N-1} \sum_{n=2}^N \left[\mathbb{E}\mathbf{t}_n\mathbf{t}_n^T - A^{new}\mathbb{E}\mathbf{t}_{n-1}\mathbf{t}_n^T - \mathbb{E}\mathbf{t}_n\mathbf{t}_{n-1}^T(A^{new})^T + A^{new}\mathbb{E}\mathbf{t}_{n-1}\mathbf{t}_{n-1}^T(A^{new})^T \right].\end{aligned}$$

Здесь $\mathbb{E}\mathbf{t}_n\mathbf{t}_{n-1}^T = J_{n-1}\hat{V}_n + \mathbb{E}\mathbf{t}_n\mathbb{E}\mathbf{t}_{n-1}^T$.

6. Пример применения фильтра Калмана: задача стабилизации ориентации объекта.

Рассмотрим пример задачи стабилизации ориентации объекта в абсолютном пространстве. Такая задача часто встречается на практике, например, при стабилизации дула танка во время движения¹ или стабилизации направления взора видео-, кино- и тепловизионной камеры, установленной на подвижном основании².

В большинстве случаев для решения данной задачи на объект устанавливают датчики абсолютной угловой скорости (MEMS, оптоволоконные и др.), которые используются для оценки ориентации объекта. Непосредственно ориентация объекта может быть вычислена путем интегрирования угловых скоростей. Однако, эти датчики имеют существенный недостаток, связанный с тем, что в их показаниях присутствует случайная низкочастотная составляющая, называемая дрейфом. При интегрировании дрейф приводит к тому, что со временем показания датчика начинают меняться даже при отсутствии движения. Показания датчика можно представить в виде:

$$\boldsymbol{\omega} = \tilde{\boldsymbol{\omega}} + \boldsymbol{\nu},$$

где $\boldsymbol{\omega}$ — истинная проекция угловой скорости на ось чувствительности датчика, $\boldsymbol{\nu}$ — составляющая дрейфа. Дрейф обычно достаточно хорошо моделируется броуновским движением:

$$\dot{\boldsymbol{\nu}} = \boldsymbol{\varepsilon}_\nu,$$

где $\boldsymbol{\varepsilon}_\nu$ — нормальный шум. Дисперсия $\boldsymbol{\varepsilon}_\nu$ является заводской характеристикой датчика и предполагается известной.

Для компенсации дрейфа на объект наряду с датчиками угловых скоростей устанавливают акселерометры. Показания акселерометра можно моделировать так:

$$\tilde{\mathbf{a}} = \mathbf{A}\mathbf{g} + \mathbf{a} + \boldsymbol{\varepsilon}_a,$$

где \mathbf{g} — ускорение свободного падения, \mathbf{a} — истинное ускорение, $\boldsymbol{\varepsilon}_a$ — нормальный шум, дисперсия которого, как правило, полагается единичной. Матрица \mathbf{A} задает ориентацию объекта в поле силы тяжести.

Возникает задача совмещения информации от акселерометров и датчиков угловых скоростей с целью компенсации дрейфа при определении ориентации объекта. Для решения данной задачи применяют фильтр Калмана.

В целях упрощения выкладок будем рассматривать задачу определения ориентации объекта в вертикальной плоскости. В этом случае ориентация задается одним углом φ . Движение объекта описывается одним кинематическим уравнением:

$$\dot{\varphi} = \boldsymbol{\omega}.$$

Показания акселерометров имеют вид:

$$\begin{aligned}\tilde{a}_x &= g \sin \varphi + a_x + \varepsilon_x, \\ \tilde{a}_z &= -g \cos \varphi + a_z + \varepsilon_z,\end{aligned}$$

Полагая, что стабилизация происходит в окрестности горизонтального положения объекта, линеаризуем уравнения показаний акселерометров. Тогда вертикальная составляющая становится неинформативной, горизонтальная же имеет вид:

$$\tilde{a}_x = g\varphi + a_x + \varepsilon_x.$$

¹<http://www.youtube.com/watch?v=Lj75U7TkBBY>

²<http://www.robycam.ru>

Проекция истинного ускорения a_x является неизвестной, и обычно ее моделируют случайным процессом, спектральные характеристики которого отражают ускорения при эксплуатации объекта. В большинстве случаев, для моделирования спектра хватает уравнения второго порядка:

$$k_1 \ddot{a}_x + k_2 \dot{a}_x + a_x = r_x,$$

где k_1, k_2 — коэффициенты, определяющие форму спектральной характеристики, r_x — нормальный шум, чья дисперсия определяет интенсивность ожидаемых значений ускорения.

Общая система может быть записана в матричном виде:

$$\begin{aligned}\dot{\xi} &= C\xi + Br, \\ \tilde{a} &= H^T \xi + \varepsilon_x,\end{aligned}$$

где

$$\xi = \begin{pmatrix} \varphi \\ \nu \\ a_x \\ \dot{a}_x \end{pmatrix}, \quad r = \begin{pmatrix} \tilde{\omega} \\ \varepsilon_\nu \\ r_x \end{pmatrix}, \quad C = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -k_2/k_1 & -1/k_1 \end{pmatrix},$$

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad H = \begin{pmatrix} g \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Для данной системы может быть построен фильтр Калмана, доставляющий оценку φ . На этом принципе основаны алгоритмы оценок в ряде систем, в частности, в существующей системе для съемок на подвижном основании RobyCam³.

7. Ограничения фильтра Калмана. Расширенный фильтр Калмана.

Фильтр Калмана выводится в предположениях нормальных линейных моделей (4). В реальной ситуации сигналы зачастую имеют нелинейную динамику и ненормальный шум. Рассмотрим, в каких ситуациях идеи фильтра Калмана могут быть обобщены на более сложные случаи.

Необходимым условием для существования точного алгоритма вывода в графической модели на рис. 2 является возможность аналитического вычисления интегралов в выражениях (7) и (8). Другим требованием к алгоритму является «неусложнение» модели для $\hat{a}(t_n)$ и $\hat{\beta}(t_n)$ с ростом n . Пусть, например, модель сенсора $p(\mathbf{x}_n | t_n)$ представляет собой смесь из K нормальных распределений, а остальные атомарные распределения в ЛДС являются нормальными линейными. Тогда $\hat{a}(t_1)$ является гауссианой, $\hat{a}(t_2)$ — смесью из K гауссиан, $\hat{a}(t_3)$ — смесью из K^2 гауссиан и т.д. Таким образом, здесь приходится иметь дело с экспоненциальным количеством слагаемых, что не позволяет реализовать метод на компьютере. В результате точный алгоритм фильтрации типа Калмана возможен только при наличии линейного перехода между мат.ожиданиями и моделью шума из т.н. экспоненциального семейства распределений.

Рассмотрим задачу нелинейной фильтрации с гауссовскими шумами:

$$\begin{aligned}t_n &= \mathbf{f}(t_{n-1}) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \Gamma), \\ \mathbf{x}_n &= \mathbf{g}(t_n) + \nu, \quad \nu \sim \mathcal{N}(0, \Sigma).\end{aligned}$$

Здесь \mathbf{f} и \mathbf{g} — известные вектор-функции.

Для такой задачи можно построить приближенный алгоритм фильтрации в реальном времени. Пусть в момент времени $n-1$ найдено текущее распределение $p(t_{n-1} | \mathbf{x}_1, \dots, \mathbf{x}_{n-1})$ вида $\mathcal{N}(t_{n-1} | \mu_{n-1}, V_{n-1})$. Приближим вектор-функцию \mathbf{f} линейной функцией в окрестности точки μ_{n-1} :

$$t_n \simeq \mathbf{f}(\mu_{n-1}) + \nabla \mathbf{f}(\mu_{n-1}) t_{n-1} + \varepsilon.$$

Тогда мы можем осуществить прогноз для t_n по формулам фильтра Калмана (5), где в качестве матрицы A выступает матрица $\nabla \mathbf{f}(\mu_{n-1})$:

$$p(t_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1}) = \mathcal{N}(t_n | \tilde{\mu}_n, \tilde{V}_n).$$

Аналогично, приближим вектор-функцию \mathbf{g} линейной функцией в окрестности точки $\tilde{\mu}_n$:

$$\mathbf{x}_n \simeq \mathbf{g}(\tilde{\mu}_n) + \nabla \mathbf{g}(\tilde{\mu}_n) t_n + \nu.$$

³<http://www.robyscam.ru/ru/video.html>

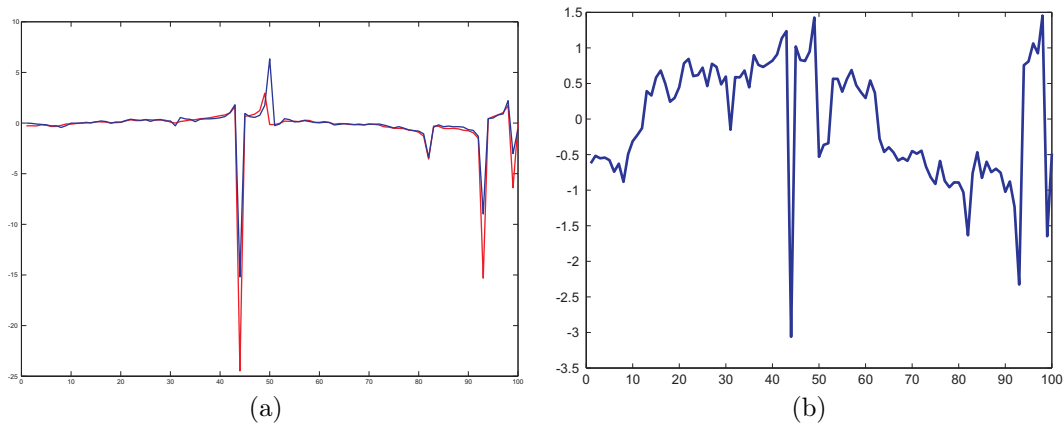


Рис. 4: Пример применения расширенного фильтра Калмана для нелинейной фильтрации сигналов. На рис. а показана истинная скрытая переменная (синяя кривая) и восстановленная скрытая переменная (красная кривая). На рис. б показан наблюдаемый сигнал.

Тогда мы можем провести коррекцию по формулам (6), где в качестве матрицы C выступает $\nabla g(\tilde{\mu}_n)$. Такой алгоритм фильтрации получил название расширенного фильтра Калмана. В том случае, если дисперсии шумов не слишком велики (т.е. линейная аппроксимация является адекватной), применение расширенного фильтра Калмана дает решение задачи с высокой точностью (см. пример на рис. 4).

В том случае, когда шумы не являются гауссовскими, расширенный фильтр Калмана применять нельзя. В этом случае обычно применяют фильтр частиц, в котором используются численные методы взятия интегралов на основе методов Монте Карло с марковскими цепями.