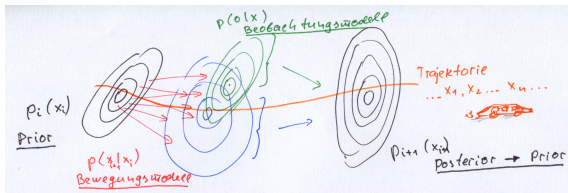


Computer Vision: Kalman Filter

D. Schlesinger – TUD/INF/KI/IS

Bayesscher Filter



Ein Objekt kann sich in einem **Zustand** $x \in \mathcal{X}$ befinden.

Zum Zeitpunkt i sei die Wahrscheinlichkeitsverteilung $p_i(x_i)$ bekannt (Achtung!!! Nicht der Zustand x_i , sondern Wahrscheinlichkeitsverteilung).

Gegeben sei ein **Bewegungsmodell** $p(x_{i+1}|x_i)$

Daraus ergibt sich die (a-priori) Wahrscheinlichkeitsverteilung der Zustände x_{i+1} als
 $p_{i+1}(x_{i+1}) = \sum_{x_i} p_i(x_i) \cdot p(x_{i+1}|x_i)$ – **Prediction**

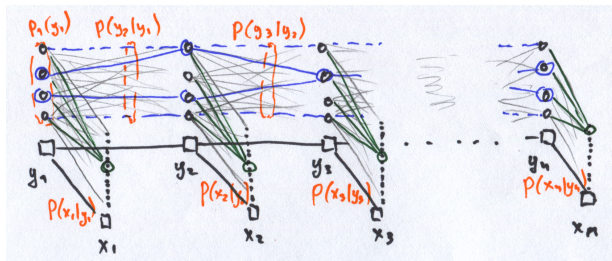
Gegeben sei ein **Beobachtungsmodell** $p(o_{i+1}|x_{i+1})$

Die Wahrscheinlichkeitsverteilung im $i+1$ Zeitpunkt ergibt sich als
 $p_{i+1}(x_{i+1}) \equiv p_{i+1}(x_{i+1}|o_{i+1}) \sim p_{i+1}(x_{i+1}) \cdot p(o_{i+1}|x_{i+1})$ – **Correction**

→ dient als a-priori Wahrscheinlichkeitsverteilung für den nächsten Schritt.

Markovsche Ketten

Die Zustandsmenge $\mathcal{X}$ ist diskret.



- + ziemlich allgemein, da „beliebige“ diskrete Modelle repräsentiert werden können
- für große Zustandsräume meist nicht geeignet (wegen Zeitkomplexität)

„Markovsche Ketten mit kontinuierlicher Zustandsmenge“ ...

[Kalman, 1960]

Zustand ist ein Vektor $x \in \mathbb{R}^n$, Wahrscheinlichkeitsverteilungen – Gaussiane.

Bewegungsmodell und Beobachtungsmodell sind linear, mit weißem Rauschen, d.h.

$$\begin{aligned}x_{i+1} &= A \cdot x_i + \epsilon \\o_i &= B \cdot x_i + \delta\end{aligned}$$

A und B sind $n \times n$ Matrizen, Gaussisch verteilte statistische Variablen (Störungen) $\epsilon = \mathcal{N}(\mu, \sigma) \sim \exp(-\|\sigma - \mu\|^2 / 2\sigma^2)$ und $\delta = \mathcal{N}(\mu', \sigma')$ (analog). Sehr oft $\mu = \mu' = 0$.

Beispiel: Zustand $x = [x, y, v_x, v_y]$ beschreibt die Lage (x, y) und die Geschwindigkeit (v_x, v_y) eines Objektes im $\mathbb{R}^2$.

Es gilt für „fast“ gleichförmige Bewegung (Δt ist der Zeitschritt):

$$\begin{aligned}x_{i+1} &= x_i + \Delta t \cdot v_{x,i} + O(\Delta t^2) \\y_{i+1} &= y_i + \Delta t \cdot v_{y,i} + O(\Delta t^2) \\v_{x,i+1} &= v_{x,i} + O(\Delta t) \\v_{y,i+1} &= v_{y,i} + O(\Delta t)\end{aligned}$$

Der Zustand im $i + 1$ Zeitpunkt ist eine lineare Kombination mit „Störungen“ $O(\cdot)$

In der Matrixform:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ v_{x,i+1} \\ v_{y,i+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ v_{x,i} \\ v_{y,i} \end{bmatrix} + \epsilon$$

Für die Beobachtung analog:

$$\begin{bmatrix} o_{x,i+1} \\ o_{y,i+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ v_{x,i} \\ v_{y,i} \end{bmatrix} + \delta$$

(nur die Lage wird beobachtet).

2D $\rightarrow$ 3D ($\mathbb{R}^6$) $\rightarrow$ 6D (mit Winkeln) ($\mathbb{R}^{12}$) $\rightarrow$...

In etwas anderer Schreibweise (Wahrscheinlichkeiten explizit) sieht man:

- 1) Prediction ist eine Faltung zweier Gaussiane

$$\begin{aligned} p_{i+1}(x_{i+1}) &= \int p_i(x_i) \cdot p(x_{i+1}|x_i) dx_i \sim \\ &\sim \int \exp\left[-\frac{\|x_i - \bar{x}\|^2}{2\sigma^2}\right] \cdot \exp\left[-\frac{\|x_{i+1} - Ax_i\|^2}{2\sigma'^2}\right] dx_i \end{aligned}$$

⇒ Ergebnis – wieder ein Gaussian.

- 2) Correction ist komponentenweise Multiplikation zweier Gaussiane

$$\begin{aligned} p_{i+1}(x_{i+1}|o_{i+1}) &= p_{i+1}(x_{i+1}) \cdot p(o_{i+1}|x_{i+1}) \sim \\ &\sim \exp\left[-\frac{\|x_{i+1} - \bar{x}_{i+1}\|^2}{2\sigma^2}\right] \cdot \exp\left[-\frac{\|o_{i+1} - Bx_{i+1}\|^2}{2\sigma'^2}\right] \end{aligned}$$

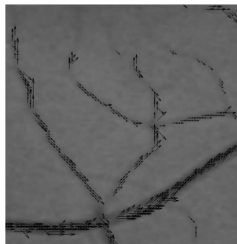
⇒ Ergebnis – wieder ein Gaussian.

Die Wahrscheinlichkeitsverteilungen werden nicht explizit neu berechnet, sondern die **Parameter** (Mittelwerte und Kovariationsmatrizen) werden propagiert.

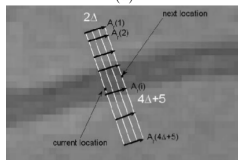
Kalman Filter – Anwendungsbeispiel

[Yedidya, Hartley, 2008]

„Verfolgung“ von Blutgefäßen:



(a)

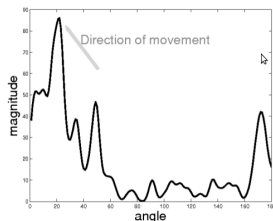


(b)

Gesucht wird eine Trajektorie

Ein Objekt bewegt sich entlang einer Bahn (Blutgefäß) und wird dabei verfolgt

„Zustand“ beschreibt Position, Geschwindigkeit, Dicke, unterwegs gesehene Grauwerte usw.



Hauptnachteil des Kalman Filters:

alles sind Gaussiane – für komplizierten Zustandsräume ungeeignet.

Besser – Mischungen von Gaussianen $p(x) = \sum_k w_k \mathcal{N}(x, \mu_k, \sigma_k)$.

Das Problem – die Anzahl der Komponenten wächst (selbst bei linearen Modellen)

$\sum_i \cdot \sum_j = \sum_{ij}$ – kann nicht parametrisch propagiert werden.

Ausweg: man approximiere ständig durch Mischung mit konstanter Anzahl der Komponenten.

Allgemeinere Fälle – „sehr viele Komponenten“, nicht linear, d.h. Ax wird zu $a(x)$ und Bx wird zu $b(x)$

man approximiere die Berechnungen durch Sampling:

- würfele x' entsprechend $p_i(x_i)$
- propagiere $x'' = a(x')$
- berechne $p(o_{i+1}|x'')$
- entscheide, ob x'' in die Lernstichprobe aufgenommen wird (accept/reject)

→ so generierte Lernstichprobe von x'' gehorcht $p_{i+1}(x_{i+1})$

Manchmal ist es gar nicht nötig, die Wahrscheinlichkeitsverteilung zu spezifizieren – gearbeitet wird immer mit den Samples.

CONDENSATION (Conditional Density Propagation) – eine bestimmte Art davon.