Lecture 3: Bayesian Filtering Equations and Kalman Filter

Simo Särkkä

January 23, 2019

Learning Outcomes

Summary of the Last Lecture

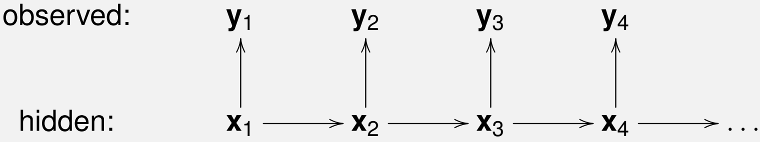
- Linear regression problem can be solved as batch problem or recursively – the latter solution is a special case of Kalman filter.
- A generic Bayesian estimation problem can also be solved as batch problem or recursively.
- If we let the linear regression parameter change between the measurements, we get a simple linear state space model – again solvable with Kalman filtering model.
- By generalizing this idea and the solution we get the Kalman filter algorithm.
- By further generalizing to non-Gaussian models results in generic probabilistic state space models.
- Bayesian filtering and smoothing methods solve Bayesian inference problems on state space models recursively.

Probabilistics State Space Models: General Model

- General probabilistic state space model:

$$
\begin{aligned}
\text{dynamic model: } \mathbf{x}_k &\sim p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) \\
\text{measurement model: } \mathbf{y}_k &\sim p(\mathbf{y}_k \,|\, \mathbf{x}_k)
\end{aligned}
$$

- $\mathbf{x}_k = (x_{k1}, \ldots, x_{kn})$ is the state and $\mathbf{y}_k = (y_{k1}, \ldots, y_{km})$ is the measurement.
- Has the form of hidden Markov model (HMM)



- *Note that HMM often refers to models with discrete state – but even with continuous state, the model is Markov and hidden . . . and thus HMM.*

Probabilistics State Space Models: Example

Example (Gaussian random walk)

Gaussian random walk model can be written as

$$
\begin{aligned}
x_k &= x_{k-1} + w_{k-1}, & w_{k-1} &\sim \mathrm{N}(0, q) \\
y_k &= x_k + e_k, & e_k &\sim \mathrm{N}(0, r),
\end{aligned}
$$

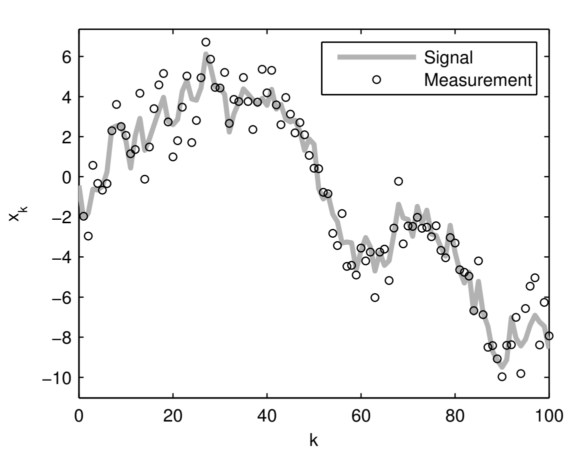
where x_k is the hidden state and y_k is the measurement. In terms of probability densities the model can be written as

$$
\begin{aligned}
p(x_k \mid x_{k-1}) &= \frac{1}{\sqrt{2\pi q}} \exp\left(-\frac{1}{2q}(x_k - x_{k-1})^2\right) \\
p(y_k \mid x_k) &= \frac{1}{\sqrt{2\pi r}} \exp\left(-\frac{1}{2r}(y_k - x_k)^2\right)
\end{aligned}
$$

which is a discrete-time state space model.

Probabilistics State Space Models: Example (cont.)

Example (Gaussian random walk (cont.))



Probabilistics State Space Models: Further Examples

- Linear Gauss-Markov model:

$$\mathbf{x}_k = \mathbf{A}_{k-1}\,\mathbf{x}_{k-1} + \mathbf{q}_{k-1}$$
$$\mathbf{y}_k = \mathbf{H}_k\,\mathbf{x}_k + \mathbf{r}_k,$$

- Gaussian driven non-linear model:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{q}_{k-1})$$
$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k).$$

- Hierarchical and/or non-Gaussian models

$$\mathbf{q}_{k-1} \sim \mathrm{Dirichlet}(\mathbf{q}_{k-1} \mid \boldsymbol{\alpha})$$
$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{q}_{k-1})$$
$$\sigma_k^2 \sim \mathrm{InvGamma}(\sigma_k^2 \mid \sigma_{k-1}^2, \gamma)$$
$$\mathbf{r}_k \sim \mathrm{N}(\mathbf{0}, \sigma_k^2\,\mathbf{I})$$
$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k).$$

Probabilistics State Space Models: Markov and Independence Assumptions

- The dynamic model $p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})$ is Markovian:
  1. Future $\mathbf{x}_k$ is independent of the past given the present (here "present" is $\mathbf{x}_{k-1}$):
  $$p(\mathbf{x}_k \,|\, \mathbf{x}_{1:k-1}, \mathbf{y}_{1:k-1}) = p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}).$$
  2. Past $\mathbf{x}_{k-1}$ is independent of the future given the present (here "present" is $\mathbf{x}_k$):
  $$p(\mathbf{x}_{k-1} \,|\, \mathbf{x}_{k:T}, \mathbf{y}_{k:T}) = p(\mathbf{x}_{k-1} \,|\, \mathbf{x}_k).$$
- The measurements $\mathbf{y}_k$ are conditionally independent given $\mathbf{x}_k$:
$$p(\mathbf{y}_k \,|\, \mathbf{x}_{1:k}, \mathbf{y}_{1:k-1}) = p(\mathbf{y}_k \,|\, \mathbf{x}_k).$$

Bayesian Filter: Principle

- Bayesian filter computes the distribution

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$$

- Given the following:
  1. Prior distribution $p(\mathbf{x}_0)$.
  2. State space model:

$$\begin{aligned} \mathbf{x}_k &\sim p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) \\ \mathbf{y}_k &\sim p(\mathbf{y}_k \mid \mathbf{x}_k), \end{aligned}$$

  3. Measurement sequence $\mathbf{y}_{1:k} = \mathbf{y}_1, \ldots, \mathbf{y}_k$.
- Computation is based on recursion rule for incorporation of the new measurement $\mathbf{y}_k$ into the posterior:

$$p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) \longrightarrow p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$$

Bayesian Filter: Derivation of Prediction Step

- Assume that we know the posterior distribution of previous time step:

$$p(\mathbf{x}_{k-1} \,|\, \mathbf{y}_{1:k-1}).$$

- The joint distribution of $\mathbf{x}_k$, $\mathbf{x}_{k-1}$ given $\mathbf{y}_{1:k-1}$ can be computed as (recall the Markov property):

$$\begin{aligned} p(\mathbf{x}_k, \mathbf{x}_{k-1} \,|\, \mathbf{y}_{1:k-1}) &= p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}, \mathbf{y}_{1:k-1})\, p(\mathbf{x}_{k-1} \,|\, \mathbf{y}_{1:k-1}) \\ &= p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})\, p(\mathbf{x}_{k-1} \,|\, \mathbf{y}_{1:k-1}), \end{aligned}$$

- Integrating over $\mathbf{x}_{k-1}$ gives the Chapman-Kolmogorov equation

$$p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})\, p(\mathbf{x}_{k-1} \,|\, \mathbf{y}_{1:k-1})\, \mathrm{d}\mathbf{x}_{k-1}.$$

- This is the prediction step of the Bayesian filter.

Bayesian Filter: Derivation of Update Step

- Now we have:
  1. Prior distribution from the Chapman-Kolmogorov equation
     $$p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})$$
  2. Measurement likelihood from the state space model:
     $$p(\mathbf{y}_k \mid \mathbf{x}_k)$$
- The posterior distribution can be computed by the Bayes' rule (recall the conditional independence of measurements):
$$\begin{aligned} p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) &= \frac{1}{Z_k} p(\mathbf{y}_k \mid \mathbf{x}_k, \mathbf{y}_{1:k-1}) \, p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \\ &= \frac{1}{Z_k} p(\mathbf{y}_k \mid \mathbf{x}_k) \, p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \end{aligned}$$
- This is the update step of the Bayesian filter.

Bayesian Filter: Formal Equations

Bayesian filter

- Initialization: The recursion starts from the prior distribution $p(\mathbf{x}_0)$.
- Prediction: by the Chapman-Kolmogorov equation

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k \mid \mathbf{x}_{k-1})\, p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1})\, \mathrm{d}\mathbf{x}_{k-1}.$$

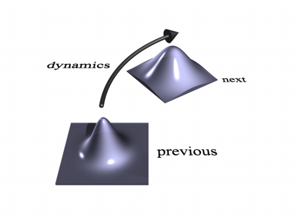
- Update: by the Bayes' rule

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) = \frac{1}{Z_k} p(\mathbf{y}_k \mid \mathbf{x}_k)\, p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}).$$

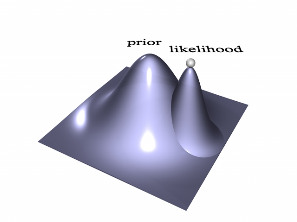
- The normalization constant $Z_k = p(\mathbf{y}_k \mid \mathbf{y}_{1:k-1})$ is given as

$$Z_k = \int p(\mathbf{y}_k \mid \mathbf{x}_k)\, p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})\, \mathrm{d}\mathbf{x}_k.$$

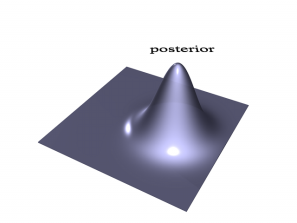
Bayesian Filter: Graphical Explanation



On prediction step the distribution of previous step is propagated through the dynamics.



Prior distribution from prediction and the likelihood of measurement.



The posterior distribution after combining the prior and likelihood by Bayes' rule.

Kalman Filter: Model

- Gaussian driven linear model, i.e., Gauss-Markov model:

$$\begin{aligned} \mathbf{x}_k &= \mathbf{A}_{k-1}\,\mathbf{x}_{k-1} + \mathbf{q}_{k-1} \\ \mathbf{y}_k &= \mathbf{H}_k\,\mathbf{x}_k + \mathbf{r}_k, \end{aligned}$$

- $\mathbf{q}_{k-1} \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}_{k-1})$ white process noise.
- $\mathbf{r}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{R}_k)$ white measurement noise.
- $\mathbf{A}_{k-1}$ is the transition matrix of the dynamic model.
- $\mathbf{H}_k$ is the measurement model matrix.
- In probabilistic terms the model is

$$\begin{aligned} p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) &= \mathrm{N}(\mathbf{x}_k \,|\, \mathbf{A}_{k-1}\,\mathbf{x}_{k-1}, \mathbf{Q}_{k-1}) \\ p(\mathbf{y}_k \,|\, \mathbf{x}_k) &= \mathrm{N}(\mathbf{y}_k \,|\, \mathbf{H}_k\,\mathbf{x}_k, \mathbf{R}_k). \end{aligned}$$

Kalman Filter: Derivation Preliminaries

- Gaussian probability density

$$\mathrm{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) = \frac{1}{(2\,\pi)^{n/2}\,|\mathbf{P}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^\mathsf{T}\,\mathbf{P}^{-1}\,(\mathbf{x}-\mathbf{m})\right),$$

- Let $\mathbf{x}$ and $\mathbf{y}$ have the Gaussian densities

$$p(\mathbf{x}) = \mathrm{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}), \qquad p(\mathbf{y} \,|\, \mathbf{x}) = \mathrm{N}(\mathbf{y} \,|\, \mathbf{H}\,\mathbf{x}, \mathbf{R}),$$

- Then the joint and marginal distributions are

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim \mathrm{N}\left(\begin{pmatrix} \mathbf{m} \\ \mathbf{H}\,\mathbf{m} \end{pmatrix}, \begin{pmatrix} \mathbf{P} & \mathbf{P}\,\mathbf{H}^\mathsf{T} \\ \mathbf{H}\,\mathbf{P} & \mathbf{H}\,\mathbf{P}\,\mathbf{H}^\mathsf{T} + \mathbf{R} \end{pmatrix} \right)$$

$$\mathbf{y} \sim \mathrm{N}(\mathbf{H}\,\mathbf{m}, \mathbf{H}\,\mathbf{P}\,\mathbf{H}^\mathsf{T} + \mathbf{R}).$$

Kalman Filter: Derivation Preliminaries (cont.)

- If the random variables $\mathbf{x}$ and $\mathbf{y}$ have the joint Gaussian probability density

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim \mathrm{N}\left(\begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix}, \begin{pmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^\mathsf{T} & \mathbf{B} \end{pmatrix} \right),$$

- Then the marginal and conditional densities of $\mathbf{x}$ and $\mathbf{y}$ are given as follows:

$$\begin{aligned} \mathbf{x} &\sim \mathrm{N}(\mathbf{a}, \mathbf{A}) \\ \mathbf{y} &\sim \mathrm{N}(\mathbf{b}, \mathbf{B}) \\ \mathbf{x} \mid \mathbf{y} &\sim \mathrm{N}(\mathbf{a} + \mathbf{C}\,\mathbf{B}^{-1}\,(\mathbf{y} - \mathbf{b}), \mathbf{A} - \mathbf{C}\,\mathbf{B}^{-1}\mathbf{C}^\mathsf{T}) \\ \mathbf{y} \mid \mathbf{x} &\sim \mathrm{N}(\mathbf{b} + \mathbf{C}^\mathsf{T}\,\mathbf{A}^{-1}\,(\mathbf{x} - \mathbf{a}), \mathbf{B} - \mathbf{C}^\mathsf{T}\,\mathbf{A}^{-1}\,\mathbf{C}). \end{aligned}$$

Kalman Filter: Derivation of Prediction Step

- Assume that the posterior distribution of previous step is Gaussian

$$p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) = \mathrm{N}(\mathbf{x}_{k-1} \mid \mathbf{m}_{k-1}, \mathbf{P}_{k-1}).$$

- The Chapman-Kolmogorov equation now gives

$$\begin{aligned} p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) &= \int p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) \, p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) \, \mathrm{d}\mathbf{x}_{k-1} \\ &= \int \mathrm{N}(\mathbf{x}_k \mid \mathbf{A}_{k-1} \, \mathbf{x}_{k-1}, \mathbf{Q}_{k-1}) \; \mathrm{N}(\mathbf{x}_{k-1} \mid \mathbf{m}_{k-1}, \mathbf{P}_{k-1}). \end{aligned}$$

- Using the Gaussian distribution computation rules from previous slides, we get the prediction step

$$\begin{aligned} p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) &= \mathrm{N}(\mathbf{x}_k \mid \mathbf{A}_{k-1} \, \mathbf{m}_{k-1}, \mathbf{A}_{k-1} \, \mathbf{P}_{k-1} \, \mathbf{A}_{k-1}^{\mathsf{T}} + \mathbf{Q}_{k-1}) \\ &= \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k^-, \mathbf{P}_k^-) \end{aligned}$$

Kalman Filter: Derivation of Update Step

- The joint distribution of $\mathbf{y}_k$ and $\mathbf{x}_k$ is

$$
\begin{aligned}
p(\mathbf{x}_k, \mathbf{y}_k \mid \mathbf{y}_{1:k-1}) &= p(\mathbf{y}_k \mid \mathbf{x}_k)\, p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \\
&= \mathrm{N}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{y}_k \end{bmatrix} \,\middle|\, \mathbf{m}'', \mathbf{P}'' \right),
\end{aligned}
$$

where

$$
\mathbf{m}'' = \begin{pmatrix} \mathbf{m}_k^- \\ \mathbf{H}_k\, \mathbf{m}_k^- \end{pmatrix}
$$

$$
\mathbf{P}'' = \begin{pmatrix} \mathbf{P}_k^- & \mathbf{P}_k^-\, \mathbf{H}_k^\mathsf{T} \\ \mathbf{H}_k\, \mathbf{P}_k^- & \mathbf{H}_k\, \mathbf{P}_k^-\, \mathbf{H}_k^\mathsf{T} + \mathbf{R}_k \end{pmatrix}.
$$

Kalman Filter: Derivation of Update Step (cont.)

- The conditional distribution of $\mathbf{x}_k$ given $\mathbf{y}_k$ is then given as

$$\begin{aligned} p(\mathbf{x}_k \mid \mathbf{y}_k, \mathbf{y}_{1:k-1}) &= p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \\ &= \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k, \mathbf{P}_k), \end{aligned}$$

where

$$\begin{aligned} \mathbf{S}_k &= \mathbf{H}_k \, \mathbf{P}_k^- \, \mathbf{H}_k^\mathsf{T} + \mathbf{R}_k \\ \mathbf{K}_k &= \mathbf{P}_k^- \, \mathbf{H}_k^\mathsf{T} \, \mathbf{S}_k^{-1} \\ \mathbf{m}_k &= \mathbf{m}_k^- + \mathbf{K}_k \, [\mathbf{y}_k - \mathbf{H}_k \, \mathbf{m}_k^-] \\ \mathbf{P}_k &= \mathbf{P}_k^- - \mathbf{K}_k \, \mathbf{S}_k \, \mathbf{K}_k^\mathsf{T}. \end{aligned}$$

Kalman Filter: Equations

Kalman Filter

- Initialization: $\mathbf{x}_0 \sim \mathrm{N}(\mathbf{m}_0, \mathbf{P}_0)$
- Prediction step:

$$
\begin{aligned}
\mathbf{m}_k^- &= \mathbf{A}_{k-1}\,\mathbf{m}_{k-1} \\
\mathbf{P}_k^- &= \mathbf{A}_{k-1}\,\mathbf{P}_{k-1}\,\mathbf{A}_{k-1}^\mathsf{T} + \mathbf{Q}_{k-1}.
\end{aligned}
$$

- Update step:

$$
\begin{aligned}
\mathbf{v}_k &= \mathbf{y}_k - \mathbf{H}_k\,\mathbf{m}_k^- \\
\mathbf{S}_k &= \mathbf{H}_k\,\mathbf{P}_k^-\,\mathbf{H}_k^\mathsf{T} + \mathbf{R}_k \\
\mathbf{K}_k &= \mathbf{P}_k^-\,\mathbf{H}_k^\mathsf{T}\,\mathbf{S}_k^{-1} \\
\mathbf{m}_k &= \mathbf{m}_k^- + \mathbf{K}_k\,\mathbf{v}_k \\
\mathbf{P}_k &= \mathbf{P}_k^- - \mathbf{K}_k\,\mathbf{S}_k\,\mathbf{K}_k^\mathsf{T}.
\end{aligned}
$$

Kalman Filter: Properties

- Kalman filter can be applied only to linear Gaussian models, for non-linearities we need e.g. EKF or UKF.
- The covariance equation is independent of measurements – the gain sequence could be computed and stored offline.
- If the model is time-invariant, the gain converges to a constant $\mathbf{K}_k \to \mathbf{K}$ and the filter becomes stationary:

$$\mathbf{m}_k = (\mathbf{A} - \mathbf{K}\,\mathbf{H}\,\mathbf{A})\,\mathbf{m}_{k-1} + \mathbf{K}\,\mathbf{y}_k$$

- The gain of the above stationary Kalman filter can be computed as $\mathbf{K} = \mathbf{P}^-\,\mathbf{H}^\mathsf{T}\,(\mathbf{H}\,\mathbf{P}^-\,\mathbf{H}^\mathsf{T} + \mathbf{R})^{-1}$, where $\mathbf{P}^-$ is the solution to the following discrete-time algebraic Riccati equation (DARE):

$$\mathbf{P}^- = \mathbf{A}\,\mathbf{P}^-\mathbf{A}^\mathsf{T} + \mathbf{Q} - \mathbf{A}\,\mathbf{P}^-\,\mathbf{H}^\mathsf{T}\,(\mathbf{H}\,\mathbf{P}^-\,\mathbf{H}^\mathsf{T} + \mathbf{R})^{-1}\mathbf{H}\,\mathbf{P}^-\,\mathbf{A}^\mathsf{T}$$

Kalman Filter: Random Walk Example

Example (Kalman filter for Gaussian random walk)

Filtering density is Gaussian

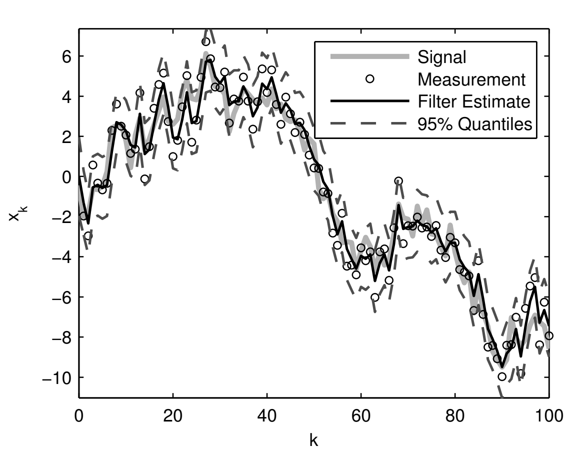
$$p(x_{k-1} \mid y_{1:k-1}) = \mathrm{N}(x_{k-1} \mid m_{k-1}, P_{k-1}).$$

The Kalman filter prediction and update equations are

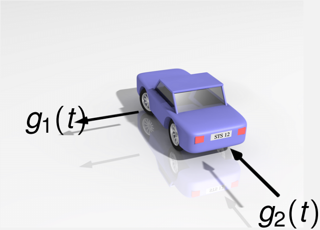
$$\begin{aligned}
m_k^- &= m_{k-1} \\
P_k^- &= P_{k-1} + q \\
m_k &= m_k^- + \frac{P_k^-}{P_k^- + r}(y_k - m_k^-) \\
P_k &= P_k^- - \frac{(P_k^-)^2}{P_k^- + r}.
\end{aligned}$$

Kalman Filter: Random Walk Example (cont.)

Example (Kalman filter for Gaussian random walk (cont.))



Dynamic Model for a Car [1/3]



- The dynamics of the car in 2d (x_1, x_2) are given by the Newton's law:

$$\mathbf{g}(t) = m\,\mathbf{a}(t),$$

where $\mathbf{a}(t)$ is the acceleration, m is the mass of the car, and $\mathbf{g}(t)$ is a vector of (unknown) forces acting the car.

- We shall now model $\mathbf{g}(t)/m$ as a 2-dimensional white noise process:

$$\begin{aligned}
\mathrm{d}^2 x_1/\mathrm{d}t^2 &= w_1(t) \\
\mathrm{d}^2 x_2/\mathrm{d}t^2 &= w_2(t).
\end{aligned}$$

Dynamic Model for a Car [2/3]

- If we define $x_3(t) = \mathrm{d}x_1/\mathrm{d}t$, $x_4(t) = \mathrm{d}x_2/\mathrm{d}t$, then the model can be written as a first order system of differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{\mathbf{F}} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}}_{\mathbf{L}} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}.$$

- In shorter matrix form:

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{F}\,\mathbf{x} + \mathbf{L}\,\mathbf{w}.$$

Dynamic Model for a Car [3/3]

- If the state of the car is measured (sampled) with sampling period Δt it suffices to consider the state of the car only at the time instances $t \in \{0, \Delta t, 2\Delta t, \ldots\}$.
- The dynamic model can be discretized, which leads to the linear difference equation model

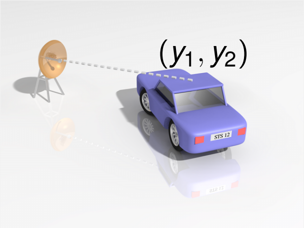
$$\begin{pmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} x_{k-1} \\ y_{k-1} \\ \dot{x}_{k-1} \\ \dot{y}_{k-1} \end{pmatrix} + \mathbf{q}_{k-1}$$

- This can also be written as

$$\mathbf{x}_k = \mathbf{A}\,\mathbf{x}_{k-1} + \mathbf{q}_{k-1},$$

where $\mathbf{x}_k = \mathbf{x}(t_k)$, $\mathbf{A}$ is the transition matrix and $\mathbf{q}_k$ is a discrete-time Gaussian noise process.

Measurement Model for a Car



- Assume that the position of the car (x_1, x_2) is measured and the measurements are corrupted by Gaussian measurement noise $e_{1,k}, e_{2,k}$:

$$
\begin{aligned}
y_{1,k} &= x_{1,k} + e_{1,k} \\
y_{2,k} &= x_{2,k} + e_{2,k}.
\end{aligned}
$$

- The measurement model can be now written as

$$
\mathbf{y}_k = \mathbf{H}\,\mathbf{x}_k + \mathbf{e}_k, \qquad \mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}
$$

Model for Car Tracking

- The dynamic and measurement models of the car now form a linear Gaussian filtering model:

$$
\begin{aligned}
\mathbf{x}_k &= \mathbf{A}\,\mathbf{x}_{k-1} + \mathbf{q}_{k-1} \\
\mathbf{y}_k &= \mathbf{H}\,\mathbf{x}_k + \mathbf{r}_k,
\end{aligned}
$$

  where $\mathbf{q}_{k-1} \sim \mathrm{N}(\mathbf{0}, \mathbf{Q})$ and $\mathbf{r}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{R})$.
- The posterior distribution is Gaussian

$$
p(\mathbf{x}_k \,|\, \mathbf{y}_1, \ldots, \mathbf{y}_k) = \mathrm{N}(\mathbf{x}_k \,|\, \mathbf{m}_k, \mathbf{P}_k).
$$

- The mean $\mathbf{m}_k$ and covariance $\mathbf{P}_k$ of the posterior distribution can be computed by the Kalman filter.

Summary

- Probabilistic state space models consist of Markovian dynamic models and conditionally independent measurement models.
- Special cases are, for example, linear Gaussian models and non-linear and non-Gaussian models.
- Bayesian filtering equations form the formal solution to general Bayesian filtering problem.
- The Bayesian filtering equations consist of prediction and update steps.
- Kalman filter is the closed form filtering solution to linear Gaussian models.

The dynamic model of the car tracking model can be written in discrete form as follows:

$$\begin{pmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} x_{k-1} \\ y_{k-1} \\ \dot{x}_{k-1} \\ \dot{y}_{k-1} \end{pmatrix} + \mathbf{q}_{k-1}$$

where $\mathbf{q}_k$ is zero mean with a covariance matrix $\mathbf{Q}$:

$$\mathbf{Q} = \begin{pmatrix} q_1^c \, \Delta t^3/3 & 0 & q_1^c \, \Delta t^2/2 & 0 \\ 0 & q_2^c \, \Delta t^3/3 & 0 & q_2^c \, \Delta t^2/2 \\ q_1^c \, \Delta t^2/2 & 0 & q_1^c \, \Delta t & 0 \\ 0 & q_2^c \, \Delta t^2/2 & 0 & q_2^c \, \Delta t \end{pmatrix}$$

Kalman Filter: Car Tracking Example [2/5]

The measurement model can be written in form

$$\mathbf{y}_k = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}_{\mathbf{H}} \begin{pmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{pmatrix} + \mathbf{e}_k,$$

where $\mathbf{e}_k$ has the covariance

$$\mathbf{R} = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix}$$

The Kalman filter prediction equations:

$$\mathbf{m}_k^- = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{m}_{k-1}$$

$$\mathbf{P}_k^- = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{P}_{k-1} \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{\mathsf{T}}$$
$$+ \begin{pmatrix} q_1^c \, \Delta t^3/3 & 0 & q_1^c \, \Delta t^2/2 & 0 \\ 0 & q_2^c \, \Delta t^3/3 & 0 & q_2^c \, \Delta t^2/2 \\ q_1^c \, \Delta t^2/2 & 0 & q_1^c \, \Delta t & 0 \\ 0 & q_2^c \, \Delta t^2/2 & 0 & q_2^c \, \Delta t \end{pmatrix}$$

Kalman Filter: Car Tracking Example [4/5]

The Kalman filter update equations:

$$\mathbf{S}_k = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{P}_k^- \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}^\mathsf{T} + \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix}$$

$$\mathbf{K}_k = \mathbf{P}_k^- \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}^\mathsf{T} \mathbf{S}_k^{-1}$$

$$\mathbf{m}_k = \mathbf{m}_k^- + \mathbf{K}_k \left(\mathbf{y}_k - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{m}_k^- \right)$$

$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\mathsf{T}$$

Kalman Filter: Car Tracking Example [5/5]

[Kalman filter for car tracking model in Matlab]