Fall 2021 ME424 Modern Control and Estimation

Lecture Note 7: Kalman Filter - Derivations and Algorithm

Prof. Wei Zhang
Department of Mechanical and Energy Engineering
SUSTech Institute of Robotics
Southern University of Science and Technology

zhangw3@sustech.edu.cn
https://www.wzhanglab.site/

Kalman Filer Preview:

- Given stochastic linear system described by

$$x_{k+1} = A_k x_k + B_k u_k + w_k$$
$$y_k = C_k x_k + D_k u_k + v_k$$

- **Kalman filter**: compute the "best" estimate of x_k given input-output data history $\{u_j, y_j\}_{j=0}^{k}$

- **Kalman Filter Solution**: $\hat{x}_k = \boldsymbol{E}(\boldsymbol{x_k} | \boldsymbol{y_0}, \boldsymbol{y_1}, \dots, \boldsymbol{y_k})$

- **Our goal: in-depth understanding of the assumptions, derivations of Kalman filter**

Outline

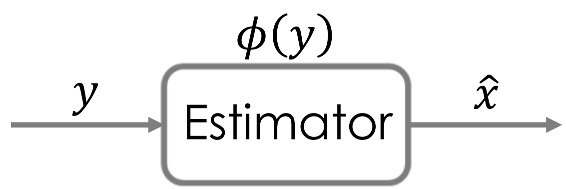
- **Minimum Mean Squared Estimation (MMSE)**
- Gaussian Random Vectors
- Kalman Filter Derivations
- Summary and Implementation

Fundamental Theorem of Estimation

- Suppose we want to estimate the value of a hidden random vector $X \in \mathrm{R}^n$ based on observations of a related vector $Y \in \mathrm{R}^m$.

- We have to know the relationship between X and Y. Suppose we take probabilistic viewpoint of their relations, namely, $(X, Y) \sim f_{XY}(x, y)$

- An estimator $\phi(y)$ is a function that maps each measurement $\mathrm{Y} = y$ to an estimate $\hat{x}$



- Mean-squared error of an estimator: $\boldsymbol{E}\left(\left|\left|\boldsymbol{\phi}(\boldsymbol{Y}) - \boldsymbol{X}\right|\right|^2\right)$

- **Example**: Given X,Y joint distribution, compute the mean-squared error for the estimator:

$$\phi(y) = 2y$$

		X = 2	3
Y	1	0.4	0.1
	2	0.2	0.3

- **Theorem: The Minimum Mean-Squared Estimator for X given $Y=y$, that minimizes $E\left(||\phi(Y)-X||^2\right)$ is given by**

$$\hat{X}_{MMSE}=\phi_{MMSE}(y)=E(X|Y=y)$$

Proof : $X\in R^n, Y\in R^m, \phi: \mathrm{R}^m\rightarrow R^n$, need to solve $\min_{\phi(\cdot)} E\left(||\phi(Y)-X||^2\right)$

Note: $E\left(||\phi(Y)-X||^2\right)=\int E\left(||\phi(Y)-X||^2 | \, Y=y\right) f_Y(y)dy$, thus we just need to find the estimator $\phi(\cdot)$ to minimize $E\left(||\phi(Y)-X||^2 | \, Y=y\right)$ for each y

$$E\left(||\phi(Y)-X||^2 |Y=y\right)=E\left((\phi(Y)-X)^T\,(\phi(Y)-X)|Y=y\right)$$
$$=E\left(\phi(Y)^T\phi(Y)-\phi(Y)^T X-X^T\phi(Y)+X^T X|Y=y\right)$$
$$=E\left(\phi(Y)^T\phi(Y)|Y=y\right)-E\left(\phi(Y)^T X|Y=y\right)-E\left(X^T\phi(Y)|Y=y\right)+E(X^T X|Y=y)$$
$$=\phi(y)^T\phi(y)-2\phi(y)^T E(X|Y=y)+E\left(X^T X|Y=y\right)$$
$$=\left(\phi(y)-E(X|Y=y)\right)^T\left(\phi(y)-E(X|Y=y)\right)-E(X|Y=y)^T E(X|Y=y)+E\left(X^T X|Y=y\right)$$

⇨ Optimal ϕ is thus given by: $\phi(y)=E(X|Y=y)$

- **Remarks:**
  - 1. The MMSE is just the conditional mean !!
  - 2. To compute the MMSE, the general way is to compute the conditional mean directly
  - 3. **Important special case:**

If $(\boldsymbol{X}, \boldsymbol{Y})$ are jointly Gaussian random vectors, then there is a simple analytical form for the conditional mean

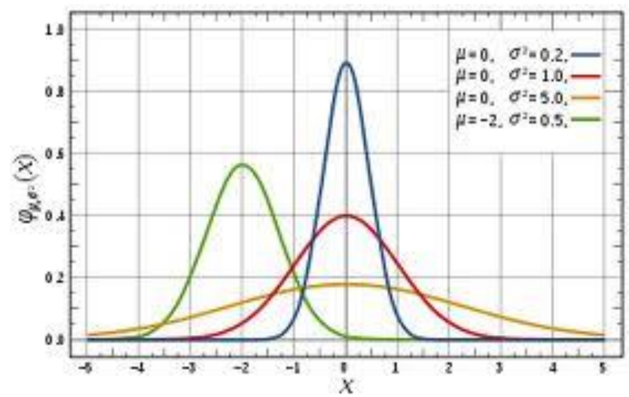
Outline

- Minimum Mean Squared Estimation (MMSE)
- **Gaussian Random Vectors**
- Kalman Filter Derivations
- Summary and Implementation

Gaussian Random Vectors

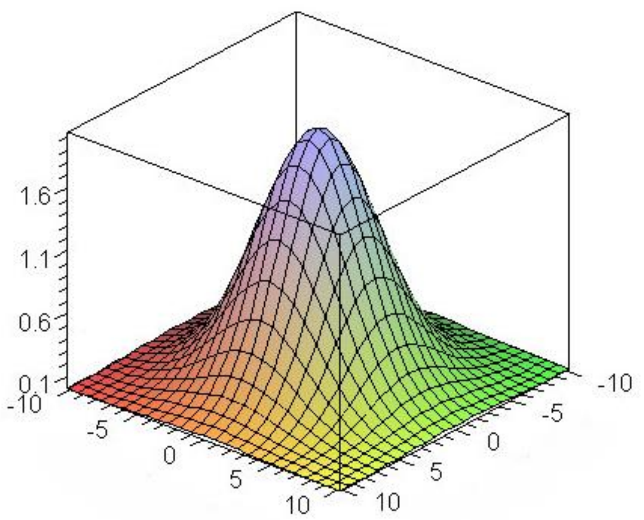
- Important due to central limit theorem
- 1D Gaussian: $\boldsymbol{X} \sim \boldsymbol{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}), \mu \in R, \sigma \in R_+$

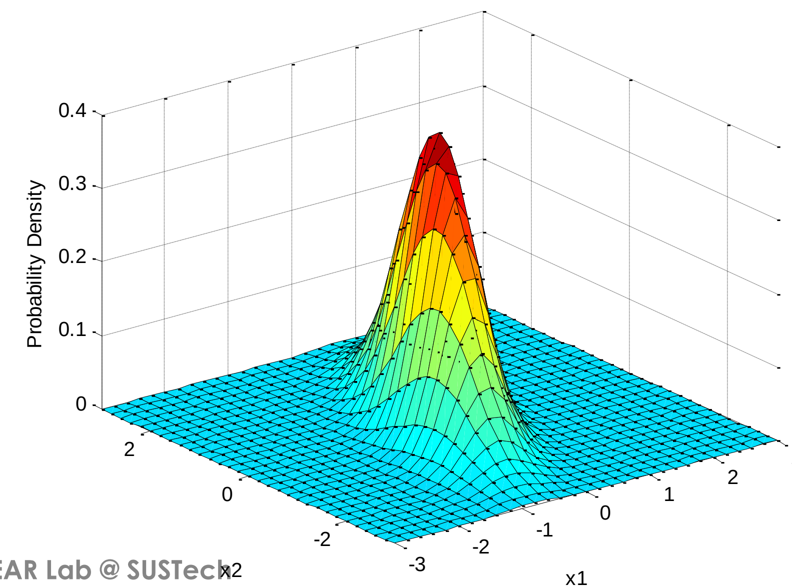
pdf : $$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



- n-D Gaussian: $\boldsymbol{X} \sim \boldsymbol{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \mu \in R^n, \Sigma \in R^{n\times n}$

pdf: $$f_X(x) = \frac{1}{(2\pi)^{\frac{n}{2}}(\det(\Sigma))^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\}$$





- Gaussian random vectors have nice properties
  - It can be presented by two parameters:

    mean vector and covariance matrix

  - **Fact 1: Uncorrelated jointly Gaussian vectors are independent**

  - **Fact 2: Linear transformation of Gaussian random vectors are Gaussian**

  - **Fact 3: Conditional Gaussian is Gaussian**

    - If general, checking whether a random variable is Gaussian or not requires computing its probability density function to see whether it is of the form of Gaussian. This can be quite involved.

- **Fact 1: Independence between two Gaussians:**
  - If $X \in R^n, Y \in R^m$ are jointly Gaussians, then $X \perp Y$ if and only if

$$E\left(XY^T\right) = E(X) \cdot E(Y)^T$$

$$Cov(X, Y) = 0$$

  - However, if X, Y are both Gaussians, but are not jointly Gaussian, then the above tests do not hold in general
    - See supplemental note on joint Gaussian random vectors

- **Fact 2: Affine transformation of Gaussian is still Gaussian**

Let $\boldsymbol{X} \sim \boldsymbol{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \boldsymbol{\mu} \in \boldsymbol{R}^{\boldsymbol{n}}, \boldsymbol{\Sigma} \in \boldsymbol{R}^{\boldsymbol{n}\times\boldsymbol{n}}, \boldsymbol{A} \in \boldsymbol{R}^{\boldsymbol{m}\times\boldsymbol{n}}, \boldsymbol{b} \in \boldsymbol{R}^{\boldsymbol{m}}$, then

$$\boldsymbol{Z} = \boldsymbol{AX} + \boldsymbol{b} \sim \boldsymbol{N}\left(\boldsymbol{A\mu} + \boldsymbol{b}, \boldsymbol{A\Sigma A}^{\boldsymbol{T}}\right)$$

- This can be used to test whether a random variable is Gaussian or not

Example: $X \sim N(\mu, \Sigma), X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 2 & 1 \\ -1 & 1 & 3 \end{bmatrix}, \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix},$

- Is X_2 Guassian?

- Is $Z = a_2X_2 + a_3X_3$ a Gaussian?

- Is $Y = \begin{bmatrix} X_1 \\ X_3 \end{bmatrix}$ a Gaussian?

- Is $X_2 \perp X_1$? How about X_1 and X_3

- **Fact 3: Conditional Gaussian is Gaussian**: Let $X \in R^n, Y \in R^m$ be **jointly** Gaussian with mean μ_X, μ_Y, covariance $\Sigma_X, \Sigma_Y, \Sigma_{XY}, \Sigma_{YX}$, i.e,

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_X & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_Y \end{bmatrix}\right)$$

then the conditional distribution of X given $Y = y$ is Gaussian

$$X|Y = y \sim N(\mu_{X|Y=y}, \Sigma_{X|Y=y}),$$

where

$$\mu_{X|Y=y} = \mu_X + \Sigma_{XY}\Sigma_Y^{-1}(y - \mu_Y)$$

$$\Sigma_{X|Y=y} = \Sigma_X - \Sigma_{XY}\Sigma_Y^{-1}\Sigma_{YX}$$

- **Example**: suppose $Z = \begin{bmatrix} X \\ Y \end{bmatrix}$, where $X \in R^2, Y \in R$, and $Z \sim N\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 8 \end{bmatrix}\right)$

- Another example: $Z = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \end{bmatrix}$, $Z \sim N\left(\begin{bmatrix} -1 \\ 5 \\ 4 \end{bmatrix}, \begin{bmatrix} 1 & -1 & 2 \\ -1 & 4 & 0.5 \\ 2 & 0.5 & 9 \end{bmatrix}\right)$

- MMSE example: $X \sim N\left(\begin{bmatrix}4\\4\end{bmatrix}, \begin{bmatrix}4 & 1\\1 & 2\end{bmatrix}\right)$, let $Y = \begin{bmatrix}1 & 0\\1 & 4\end{bmatrix} X + V$, where $V \sim N\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\right)$, V is independent of X. Find the MMSE of X given $Y = \begin{bmatrix}1\\1\end{bmatrix}$

- solution (continue)

Outline

- Minimum Mean Squared Estimation (MMSE)
- Gaussian Random Vectors
- **Kalman Filter Derivations**
- Summary and Implementation

Kalman Filter

- Consider a stochastic linear system described by

$$x_{k+1} = A_k x_k + B_k u_k + w_k$$
$$y_k = C_k x_k + D_k u_k + v_k$$

  - $x_k \in R^n$--- system state at time k
  - $y_k \in R^m$ --- measurement vector at time k
  - $Y_k \triangleq \left[y_0^T \;\; y_1^T \;\; \dots \;\; y_k^T\right]^T$ --- collection of measurements up to time k
  - $u_k \in \mathrm{R}^p$--- system input at time k (deterministic input)
  - $w_k \in R^n$--- process noise $\sim N(0, Q_k)$
  - $v_k \in R^p$--- measurement noise $\sim N(0, R_k)$
  - Assume $x_0 \sim N(\mu_0, \Phi_0), x_0 \perp w_k, x_0 \perp v_k, w_k \perp v_k, \forall k$

- Implications of the above assumption:
  - x_k is Gaussian and y_k is Gaussian for all $k \geq 0$ (VFY)

- **State estimation problem**: Find the MMSE of x_k given Y_k

- Solution using Fundamental Theorem: $E(x_k|Y_k)$

- **Kalman filter is just a recursive way to compute the conditional mean as new measurement comes in**

- Define: $\hat{x}_{k|k} = E(x_k|Y_k), \quad \hat{x}_{k|k-1} = E(x_k|Y_{k-1})$

$$P_{k|k} = E\left(\left(x_k - \hat{x}_{k|k}\right)\left(x_k - \hat{x}_{k|k}\right)^T \middle| \mathrm{Y_k}\right)$$

$$P_{k|k-1} = E\left(\left(x_k - \hat{x}_{k|k-1}\right)\left(x_k - \hat{x}_{k|k-1}\right)^T \middle| Y_{k-1}\right)$$

- Simplified notation: $\hat{x}_k \triangleq \hat{x}_{k|k}, \quad P_k = P_{k|k}$

- Before deriving Kalman filter, let's work on an example

Example:

$$x_{k+1} = A_k x_k + w_k$$
$$y_k = C_k x_k$$

where $x_0 \sim N(0,,\Sigma_x), w_k \sim N(0,\Sigma_w), A = \Sigma_w = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \Sigma_x = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, C = [1 \quad 2].$

Compute $\hat{x}_0 = E(x_0|y_0 = 1)$ and $\hat{x}_1 = E(x_1|y_0 = 1, y_1 = -1)$

Example continue..

Example continue…

Derivation of Kalman Filter

- Goal of Kalman Filter: obtain recursive formula

$$\begin{Bmatrix} \hat{x}_k \\ P_k \end{Bmatrix} \longrightarrow \begin{Bmatrix} \hat{x}_{k+1} \\ P_{k+1} \end{Bmatrix}$$

- We can compute $\hat{x}_0$, P_0

- Given $\hat{x}_k$, P_k, how to compute $\hat{x}_{k+1}, P_{k+1}$: divide this recursion into two stages: prediction and measurement update

- **Step 1: prediction** (try to compute $\hat{x}_{k+1|k}, P_{k+1|k}$ using $\hat{x}_k, P_k$

- Summary of the **prediction step**:

$$\hat{x}_{k+1|k} = A_k \hat{x}_k + B_k u_k,$$
$$P_{k+1|k} = A_k P_k A_k^T + Q_k$$

- **Step 2: measurement update**
  - We want $\hat{x}_{k+1} = E(x_{k+1}|Y_{k+1}) = E(x_{k+1}|Y_k, y_{k+1})$
  - Up to now, we have $\hat{x}_{k+1|k}$ and $P_{k+1|k}$, i.e. the mean and covariance of conditional random variable $x_{k+1}|Y_k$
  - How to find the mean and covariance of $x_{k+1}|\{Y_k, y_{k+1}\}$
  - Define $Z = x_{k+1}|Y_k,\ W = y_{k+1}|Y_k \Rightarrow \hat{x}_{k+1} = E(Z|W)$

- Derivation (continue)

- Complete derivation will be posted online

Outline

- Minimum Mean Squared Estimation (MMSE)
- Gaussian Random Vectors
- Kalman Filter Derivations
- **Summary and Implementation**

Kalman Filter Notations

System model

$$x_{k+1} = A_k x_k + B_k u_k + w_k$$
$$y_k = C_k x_k + D_k u_k + v_k$$

Noise model

$$w_k \sim N(0, Q_k)$$
$$v_k \sim N(0, R_k)$$

Measurement history

$$Y_k = \{y_0, y_1, \ldots, y_k\}$$

Filtered estimate

$$\hat{x}_k = \hat{x}_{k|k} = E(x_k | Y_k)$$

Filter error

$$e_k = x_k - x_k, \text{ with } E(e_k) = 0$$

Filter error covariance

$$P_k = E(e_k e_k^T) = E(e_k e_k^T \mid Y_k)$$

Predicted estimate

$$x_{k|k-1} = E(x_k | Y_{k-1})$$

Prediction error

$$e_{k|k-1} = x_k - x_{k|k-1}, \text{ with } E(e_{k|k-1}) = 0$$

Prediction error covariance

$$P_{k|k-1} = E(e_{k|k-1} e_{k|k-1}^T) = E(e_{k|k-1} e_{k|k-1}^T | Y_{k-1})$$

Some facts:

- Unbiasedness: $E(e_k) = 0 \quad E(e_{k|k-1}) = 0$
- Error covariance equals the conditional error covariance

$$E(e_k e_k^T) = E(e_k e_k^T \mid Y_k) \qquad E(e_{k|k-1} e_{k|k-1}^T) = E(e_{k|k-1} e_{k|k-1}^T | Y_{k-1})$$

- Mean squared error: $E\left(\|e_k\|^2\right) = \text{trace}(P_k),\ E\left(\|e_{k|k-1}\|^2\right) = \text{trace}(P_{k|k-1})$

Kalman Filter Diagram

$$\begin{Bmatrix} x_{k-1} \\ P_{k-1} \end{Bmatrix} \xrightarrow{\text{prediction}} \begin{Bmatrix} x_{k|k-1} \\ P_{k|k-1} \end{Bmatrix} \xrightarrow{\text{Measurement update}} \begin{Bmatrix} x_{k} \\ P_{k} \end{Bmatrix} \xrightarrow{\text{prediction}} \begin{Bmatrix} x_{k+1|k} \\ P_{k+1|k} \end{Bmatrix} \cdots\cdots$$

$\{x_{0|-1}, P_{0|-1}\}$ Initial condition

Prediction

1. Predict state ahead

$$x_{k|k-1} = A_{k-1}x_{k-1} + B_{k-1}u_{k-1}$$

2. Prediction error covariance

$$P_{k|k-1} = A_{k-1}P_{k-1}A_{k-1}^T + Q_{k-1}$$

Initial condition $\{x_0, P_0\}$

Measurement Update

1. Compute Kalman gain

$$K_k = P_{k|k-1}C_k^T\left(C_k P_{k|k-1}C_k^T + R_k\right)^{-1}$$

2. Update estimate

$$x_k = x_{k|k-1} + K_k\left(y_k - C_k x_{k|k-1} - D_k u_k\right)$$

Innovation

3. Update error covariance

$$P_k = \left(I - K_k C_k\right)P_{k|k-1}$$

$k = k+1$

Either of the two initial conditions is enough to start the iteration

Coding Example