Lecture 5: Unscented Kalman filter, Gaussian Filter, GHKF and CKF

Simo Särkkä

February 6, 2019

Learning Outcomes

- EKF and SLF can be applied to filtering models of the form

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

- EKF is based on Taylor series expansions of **f** and **h**.
  - Advantages: Simple, intuitive, computationally efficient
  - Disadvantages: Local approximation, differentiability requirements, only for Gaussian noises.

- SLF is based on statistical linearization:
  - Advantages: Global approximation, no differentiability requirements, computationally efficient
  - Disadvantages: Closed form computation of expectations, only for Gaussian noises.
  - But, there is a connection to sigma-point filters (e.g., UKF).

Linearization Based Gaussian Approximation

- Problem: Determine the mean and covariance of *y*:

$$x \sim \mathsf{N}(\mu, \sigma^2)$$
$$y = \sin(x)$$

- Linearization based approximation:

$$y = \sin(\mu) + \frac{\partial \sin(\mu)}{\partial \mu}(x - \mu) + \dots$$

which gives

$$\mathsf{E}[y] \approx \mathsf{E}[\sin(\mu) + \cos(\mu)(x - \mu)] = \sin(\mu)$$
$$\mathsf{Cov}[y] \approx \mathsf{E}[(\sin(\mu) + \cos(\mu)(x - \mu) - \sin(\mu))^2] = \cos^2(\mu)\, \sigma^2.$$

- Form 3 sigma points as follows:

$$\mathcal{X}^{(0)} = \mu$$
$$\mathcal{X}^{(1)} = \mu + \sigma$$
$$\mathcal{X}^{(2)} = \mu - \sigma.$$

- Let's select some weights $W^{(0)}, W^{(1)}, W^{(2)}$ such that the original mean and variance can be recovered by

$$\mu = \sum_i W^{(i)} \mathcal{X}^{(i)}$$
$$\sigma^2 = \sum_i W^{(i)} (\mathcal{X}^{(i)} - \mu)^2.$$

- We use the same formula for approximating the moments of $y = \sin(x)$ as follows:

$$\mu_y = \sum_i W^{(i)} \sin(\mathcal{X}^{(i)})$$

$$\sigma_y^2 = \sum_i W^{(i)} (\sin(\mathcal{X}^{(i)}) - \mu)^2.$$

- For vectors $\mathbf{x} \sim N(\mathbf{m}, \mathbf{P})$ the generalization of standard deviation σ is the Cholesky factor $\mathbf{L} = \sqrt{\mathbf{P}}$:

$$\mathbf{P} = \mathbf{L}\mathbf{L}^\mathsf{T}.$$

- The sigma points can be formed using columns of $\mathbf{L}$ (here c is a suitable positive constant):

$$\mathcal{X}^{(0)} = \mathbf{m}$$

$$\mathcal{X}^{(i)} = \mathbf{m} + c\,\mathbf{L}_i$$

$$\mathcal{X}^{(n+i)} = \mathbf{m} - c\,\mathbf{L}_i$$

- For transformation $\mathbf{y} = \mathbf{g}(\mathbf{x})$ the approximation is:

$$\boldsymbol{\mu}_y = \sum_i W^{(i)} \, \mathbf{g}(\mathcal{X}^{(i)})$$

$$\Sigma_y = \sum_i W^{(i)} \, (\mathbf{g}(\mathcal{X}^{(i)}) - \boldsymbol{\mu}_y) \, (\mathbf{g}(\mathcal{X}^{(i)}) - \boldsymbol{\mu}_y)^\mathsf{T}.$$

- It is convenient to define transformed sigma points:

$$\mathcal{Y}^{(i)} = \mathbf{g}(\mathcal{X}^{(i)})$$

- Joint moments of $\mathbf{x}$ and $\mathbf{y} = \mathbf{g}(\mathbf{x}) + \mathbf{q}$ are then approximated as

$$\mathsf{E}\left[\begin{pmatrix} \mathbf{x} \\ \mathbf{g}(\mathbf{x}) + \mathbf{q} \end{pmatrix}\right] \approx \sum_i W^{(i)} \begin{pmatrix} \mathcal{X}^{(i)} \\ \mathcal{Y}^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{m} \\ \boldsymbol{\mu}_y \end{pmatrix}$$

$$\mathsf{Cov}\left[\begin{pmatrix} \mathbf{x} \\ \mathbf{g}(\mathbf{x}) + \mathbf{q} \end{pmatrix}\right]$$

$$\approx \sum_i W^{(i)} \begin{pmatrix} (\mathcal{X}^{(i)} - \mathbf{m}) \, (\mathcal{X}^{(i)} - \mathbf{m})^\mathsf{T} & (\mathcal{X}^{(i)} - \mathbf{m}) \, (\mathcal{Y}^{(i)} - \boldsymbol{\mu}_y)^\mathsf{T} \\ (\mathcal{Y}^{(i)} - \boldsymbol{\mu}_y) \, (\mathcal{X}^{(i)} - \mathbf{m})^\mathsf{T} & (\mathcal{Y}^{(i)} - \boldsymbol{\mu}_y) \, (\mathcal{Y}^{(i)} - \boldsymbol{\mu}_y)^\mathsf{T} + \mathbf{Q} \end{pmatrix}$$

Unscented transform

The unscented transform approximation to the joint distribution of $\mathbf{x}$ and $\mathbf{y} = \mathbf{g}(\mathbf{x}) + \mathbf{q}$ where $\mathbf{x} \sim N(\mathbf{m}, \mathbf{P})$ and $\mathbf{q} \sim N(\mathbf{0}, \mathbf{Q})$ is

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{m} \\ \boldsymbol{\mu}_U \end{pmatrix}, \begin{pmatrix} \mathbf{P} & \mathbf{C}_U \\ \mathbf{C}_U^\mathsf{T} & \mathbf{S}_U \end{pmatrix} \right),$$

where the sub-matrices are formed as follows:

1. Form the sigma points as

$$\begin{aligned} \mathcal{X}^{(0)} &= \mathbf{m} \\ \mathcal{X}^{(i)} &= \mathbf{m} + \sqrt{n+\lambda} \left[\sqrt{\mathbf{P}} \right]_i \\ \mathcal{X}^{(i+n)} &= \mathbf{m} - \sqrt{n+\lambda} \left[\sqrt{\mathbf{P}} \right]_i, \quad i = 1, \ldots, n \end{aligned}$$

Unscented transform (cont.)

2. Propagate the sigma points through $\mathbf{g}(\cdot)$:

$$\mathcal{Y}^{(i)} = \mathbf{g}(\mathcal{X}^{(i)}), \quad i = 0, \dots, 2n.$$

3. The sub-matrices are then given as:

$$\boldsymbol{\mu}_U = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}^{(i)}$$

$$\mathbf{S}_U = \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{Y}^{(i)} - \boldsymbol{\mu}_U)(\mathcal{Y}^{(i)} - \boldsymbol{\mu}_U)^\mathsf{T} + \mathbf{Q}$$

$$\mathbf{C}_U = \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{X}^{(i)} - \mathbf{m})(\mathcal{Y}^{(i)} - \boldsymbol{\mu}_U)^\mathsf{T}.$$

Unscented transform (cont.)

- λ is a scaling parameter defined as $\lambda = \alpha^2 (n + \kappa) - n$.
- α and κ determine the spread of the sigma points.
- Weights $W_i^{(m)}$ and $W_i^{(c)}$ are given as follows:
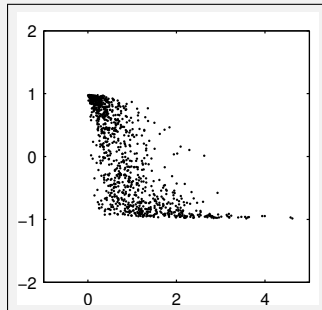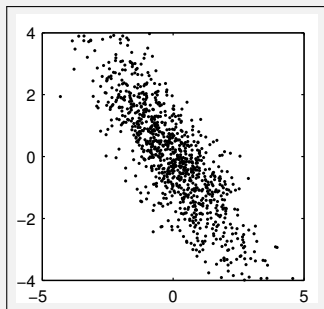
$$W_0^{(m)} = \lambda/(n + \lambda)$$
$$W_0^{(c)} = \lambda/(n + \lambda) + (1 - \alpha^2 + \beta)$$
$$W_i^{(m)} = 1/\{2(n + \lambda)\}, \quad i = 1, \ldots, 2n$$
$$W_i^{(c)} = 1/\{2(n + \lambda)\}, \quad i = 1, \ldots, 2n,$$

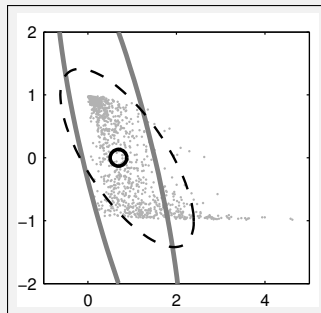- β can be used for incorporating prior information on the (non-Gaussian) distribution of **x**.

$$\implies$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \sim \mathsf{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 & -2 \\ -2 & 3 \end{pmatrix} \right)$$

$$\frac{\mathrm{d}y_1}{\mathrm{d}t} = \exp(-y_1), \quad y_1(0) = x_1$$

$$\frac{\mathrm{d}y_2}{\mathrm{d}t} = -\frac{1}{2}y_2^3, \qquad y_2(0) = x_2$$

$\Longrightarrow$
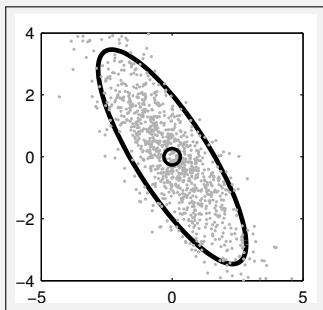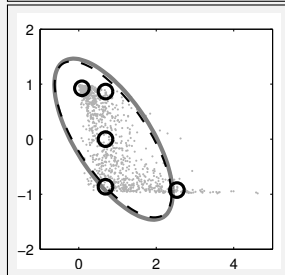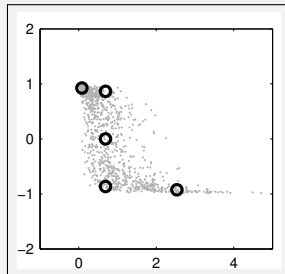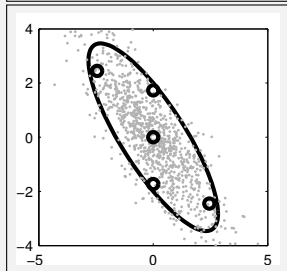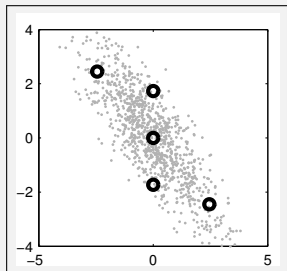
- Assume that the filtering distribution of previous step is Gaussian

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

- The joint distribution of $\mathbf{x}_{k-1}$ and $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1}$ can be approximated with UT as Gaussian

$$p(\mathbf{x}_{k-1}, \mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \approx \mathrm{N}\left(\begin{bmatrix} \mathbf{x}_{k-1} \\ \mathbf{x}_k \end{bmatrix} \;\middle|\; \begin{pmatrix} \mathbf{m}'_1 \\ \mathbf{m}'_2 \end{pmatrix}, \begin{pmatrix} \mathbf{P}'_{11} & \mathbf{P}'_{12} \\ (\mathbf{P}'_{12})^{\mathsf{T}} & \mathbf{P}'_{22} \end{pmatrix} \right),$$

as follows.

- Form the sigma points $\mathcal{X}^{(i)}$ of $\mathbf{x}_{k-1} \sim \mathrm{N}(\mathbf{m}_{k-1}, \mathbf{P}_{k-1})$ and compute the transformed sigma points as $\hat{\mathcal{X}}^{(i)} = \mathbf{f}(\mathcal{X}^{(i)})$.
- The expected values can now be expressed as:

$$\mathbf{m}'_1 = \mathbf{m}_{k-1}$$
$$\mathbf{m}'_2 = \sum_i W_i^{(m)} \, \hat{\mathcal{X}}^{(i)}$$

- The blocks of covariance can be expressed as:

$$
\begin{aligned}
\mathbf{P}'_{11} &= \mathbf{P}_{k-1} \\
\mathbf{P}'_{12} &= \sum_i W_i^{(c)} (\mathcal{X}^{(i)} - \mathbf{m}_{k-1}) (\hat{\mathcal{X}}^{(i)} - \mathbf{m}'_2)^\mathsf{T} \\
\mathbf{P}'_{22} &= \sum_i W_i^{(c)} (\hat{\mathcal{X}}^{(i)} - \mathbf{m}'_2) (\hat{\mathcal{X}}^{(i)} - \mathbf{m}'_2)^\mathsf{T} + \mathbf{Q}_{k-1}
\end{aligned}
$$

- The prediction mean and covariance of $\mathbf{x}_k$ are then $\mathbf{m}'_2$ and $\mathbf{P}'_{22}$, and thus we get

$$
\begin{aligned}
\mathbf{m}_k^- &= \sum_i W_i^{(m)} \hat{\mathcal{X}}^{(i)} \\
\mathbf{P}_k^- &= \sum_i W_i^{(c)} (\hat{\mathcal{X}}^{(i)} - \mathbf{m}_k^-) (\hat{\mathcal{X}}^{(i)} - \mathbf{m}_k^-)^\mathsf{T} + \mathbf{Q}_{k-1}
\end{aligned}
$$

- For the joint distribution of $\mathbf{x}_k$ and $\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{r}_k$ we similarly get

$$
p(\mathbf{x}_k, \mathbf{y}_k, | \mathbf{y}_{1:k-1}) \approx N\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{y}_k \end{bmatrix} \; \middle| \; \begin{pmatrix} \mathbf{m}_1'' \\ \mathbf{m}_2'' \end{pmatrix}, \begin{pmatrix} \mathbf{P}_{11}'' & \mathbf{P}_{12}'' \\ (\mathbf{P}_{12}'')^\mathsf{T} & \mathbf{P}_{22}'' \end{pmatrix} \right),
$$

- If $\mathcal{X}^{-(i)}$ are the sigma points of $\mathbf{x}_k \sim N(\mathbf{m}_k^-, \mathbf{P}_k^-)$ and $\hat{\mathcal{Y}}^{(i)} = \mathbf{h}(\mathcal{X}^{-(i)})$, we get:

$$
\mathbf{m}_1'' = \mathbf{m}_k^-
$$

$$
\mathbf{m}_2'' = \sum_i W_i^{(m)} \hat{\mathcal{Y}}^{(i)}
$$

$$
\mathbf{P}_{11}'' = \mathbf{P}_k^-
$$

$$
\mathbf{P}_{12}'' = \sum_i W_i^{(c)} (\mathcal{X}^{-(i)} - \mathbf{m}_k^-)(\hat{\mathcal{Y}}^{(i)} - \mathbf{m}_2'')^\mathsf{T}
$$

$$
\mathbf{P}_{22}'' = \sum_i W_i^{(c)} (\hat{\mathcal{Y}}^{(i)} - \mathbf{m}_2'')(\hat{\mathcal{Y}}^{(i)} - \mathbf{m}_2'')^\mathsf{T} + \mathbf{R}_k
$$

- Recall that if

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

  then

$$\mathbf{x} \,|\, \mathbf{y} \sim \mathsf{N}(\mathbf{a} + \mathbf{C}\,\mathbf{B}^{-1}\,(\mathbf{y} - \mathbf{b}), \mathbf{A} - \mathbf{C}\,\mathbf{B}^{-1}\mathbf{C}^\mathsf{T}).$$

- Thus we get the conditional mean and covariance:

$$\mathbf{m}_k = \mathbf{m}_k^- + \mathbf{P}_{12}''\,(\mathbf{P}_{22}'')^{-1}(\mathbf{y}_k - \mathbf{m}_2'')$$
$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{P}_{12}''\,(\mathbf{P}_{22}'')^{-1}\,(\mathbf{P}_{12}'')^\mathsf{T}.$$

Unscented Kalman filter: Prediction step

1. Form the sigma points:

$$
\begin{aligned}
\mathcal{X}_{k-1}^{(0)} &= \mathbf{m}_{k-1}, \\
\mathcal{X}_{k-1}^{(i)} &= \mathbf{m}_{k-1} + \sqrt{n+\lambda}\left[\sqrt{\mathbf{P}_{k-1}}\right]_i \\
\mathcal{X}_{k-1}^{(i+n)} &= \mathbf{m}_{k-1} - \sqrt{n+\lambda}\left[\sqrt{\mathbf{P}_{k-1}}\right]_i, \quad i = 1, \ldots, n.
\end{aligned}
$$

2. Propagate the sigma points through the dynamic model:

$$
\hat{\mathcal{X}}_k^{(i)} = \mathbf{f}(\mathcal{X}_{k-1}^{(i)}). \quad i = 0, \ldots, 2n.
$$

Unscented Kalman filter: Prediction step (cont.)

③ Compute the predicted mean and covariance:

$$\mathbf{m}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \, \hat{\mathcal{X}}_k^{(i)}$$

$$\mathbf{P}_k^- = \sum_{i=0}^{2n} W_i^{(c)} \, (\hat{\mathcal{X}}_k^{(i)} - \mathbf{m}_k^-) \, (\hat{\mathcal{X}}_k^{(i)} - \mathbf{m}_k^-)^\mathsf{T} + \mathbf{Q}_{k-1}.$$

Unscented Kalman filter: Update step

1. Form the sigma points:

$$
\begin{aligned}
\mathcal{X}_k^{-(0)} &= \mathbf{m}_k^-, \\
\mathcal{X}_k^{-(i)} &= \mathbf{m}_k^- + \sqrt{n + \lambda} \left[\sqrt{\mathbf{P}_k^-} \right]_i \\
\mathcal{X}_k^{-(i+n)} &= \mathbf{m}_k^- - \sqrt{n + \lambda} \left[\sqrt{\mathbf{P}_k^-} \right]_i, \quad i = 1, \ldots, n.
\end{aligned}
$$

2. Propagate sigma points through the measurement model:

$$
\hat{\mathcal{Y}}_k^{(i)} = \mathbf{h}(\mathcal{X}_k^{-(i)}), \quad i = 0, \ldots, 2n.
$$

Unscented Kalman Filter (UKF): Algorithm [4/4]

Unscented Kalman filter: Update step (cont.)

3. Compute the following:

$$\boldsymbol{\mu}_k = \sum_{i=0}^{2n} W_i^{(m)} \, \hat{\mathcal{Y}}_k^{(i)}$$

$$\mathbf{S}_k = \sum_{i=0}^{2n} W_i^{(c)} \, (\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k) \, (\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)^\mathsf{T} + \mathbf{R}_k$$

$$\mathbf{C}_k = \sum_{i=0}^{2n} W_i^{(c)} \, (\mathcal{X}_k^{-(i)} - \mathbf{m}_k^-) \, (\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)^\mathsf{T}$$

$$\mathbf{K}_k = \mathbf{C}_k \, \mathbf{S}_k^{-1}$$

$$\mathbf{m}_k = \mathbf{m}_k^- + \mathbf{K}_k \, [\mathbf{y}_k - \boldsymbol{\mu}_k]$$

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

- No closed form derivatives or expectations needed.
- Not a local approximation, but based on values on a larger area.
- Functions **f** and **h** do not need to be differentiable.
- Theoretically, captures higher order moments of distribution than linearization — the mean is correct for up to third order monomials.

- Not a truly global approximation, based on a small set of trial points.
- Does not work well with nearly singular covariances, i.e., with nearly deterministic systems.
- Requires more computations than EKF or SLF, e.g., Cholesky factorizations on every step.
- The covariance computation is exact only for linear functions.
- Can only be applied to models driven by Gaussian noises.

- Consider the transformation of **x** into **y**:

$$\mathbf{x} \sim \mathrm{N}(\mathbf{m}, \mathbf{P})$$
$$\mathbf{y} = \mathbf{g}(\mathbf{x}).$$

- Form Gaussian approximation to $(\mathbf{x}, \mathbf{y})$ by directly approximating the integrals:

$$\boldsymbol{\mu}_M = \int \mathbf{g}(\mathbf{x}) \, \mathrm{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x}$$

$$\mathbf{S}_M = \int (\mathbf{g}(\mathbf{x}) - \boldsymbol{\mu}_M) (\mathbf{g}(\mathbf{x}) - \boldsymbol{\mu}_M)^\mathsf{T} \, \mathrm{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x}$$

$$\mathbf{C}_M = \int (\mathbf{x} - \mathbf{m}) (\mathbf{g}(\mathbf{x}) - \boldsymbol{\mu}_M)^\mathsf{T} \, \mathrm{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x}.$$

Gaussian Moment Matching [2/2]

Gaussian moment matching

The moment matching based Gaussian approximation to the joint distribution of $\mathbf{x}$ and the transformed random variable $\mathbf{y} = \mathbf{g}(\mathbf{x}) + \mathbf{q}$ where $\mathbf{x} \sim \mathsf{N}(\mathbf{m}, \mathbf{P})$ and $\mathbf{q} \sim \mathsf{N}(\mathbf{0}, \mathbf{Q})$ is given as

$$
\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim \mathsf{N} \left(\begin{pmatrix} \mathbf{m} \\ \boldsymbol{\mu}_M \end{pmatrix}, \begin{pmatrix} \mathbf{P} & \mathbf{C}_M \\ \mathbf{C}_M^\mathsf{T} & \mathbf{S}_M \end{pmatrix} \right),
$$

where

$$
\boldsymbol{\mu}_M = \int \mathbf{g}(\mathbf{x}) \, \mathsf{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x}
$$

$$
\mathbf{S}_M = \int (\mathbf{g}(\mathbf{x}) - \boldsymbol{\mu}_M) (\mathbf{g}(\mathbf{x}) - \boldsymbol{\mu}_M)^\mathsf{T} \, \mathsf{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x} + \mathbf{Q}
$$

$$
\mathbf{C}_M = \int (\mathbf{x} - \mathbf{m}) (\mathbf{g}(\mathbf{x}) - \boldsymbol{\mu}_M)^\mathsf{T} \, \mathsf{N}(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x}.
$$

- An alternative way of writing the moment matching:

$$\boldsymbol{\mu}_M = \mathsf{E}[\mathbf{g}(\mathbf{x})]$$
$$\mathbf{S}_M = \mathrm{Cov}[\mathbf{g}(\mathbf{x})]$$
$$\mathbf{C}_M = \mathsf{E}[\mathbf{g}(\mathbf{x})\,\delta\mathbf{x}^\mathsf{T}]^\mathsf{T}.$$

where $\delta\mathbf{x} = \mathbf{x} - \mathbf{m}$.

- Differs from statistical linearization only in the **S**-terms:

$$\boldsymbol{\mu}_S = \mathsf{E}[\mathbf{g}(\mathbf{x})]$$
$$\mathbf{S}_S = \mathsf{E}[\mathbf{g}(\mathbf{x})\,\delta\mathbf{x}^\mathsf{T}]\,\mathbf{P}^{-1}\,\mathsf{E}[\mathbf{g}(\mathbf{x})\,\delta\mathbf{x}^\mathsf{T}]^\mathsf{T} + \mathbf{Q}$$
$$\mathbf{C}_S = \mathsf{E}[\mathbf{g}(\mathbf{x})\,\delta\mathbf{x}^\mathsf{T}]^\mathsf{T}.$$

- The Gaussian moment matching is equivalent to so called statistical linear regression.

Gaussian filter prediction

Compute the following Gaussian integrals:

$$\mathbf{m}_k^- = \int \mathbf{f}(\mathbf{x}_{k-1})\, N(\mathbf{x}_{k-1} \mid \mathbf{m}_{k-1}, \mathbf{P}_{k-1})\, d\mathbf{x}_{k-1}$$

$$\mathbf{P}_k^- = \int (\mathbf{f}(\mathbf{x}_{k-1}) - \mathbf{m}_k^-)\,(\mathbf{f}(\mathbf{x}_{k-1}) - \mathbf{m}_k^-)^\mathsf{T}$$
$$\times\, N(\mathbf{x}_{k-1} \mid \mathbf{m}_{k-1}, \mathbf{P}_{k-1})\, d\mathbf{x}_{k-1} + \mathbf{Q}_{k-1}.$$

Gaussian filter update

1. Compute the following Gaussian integrals:

$$\boldsymbol{\mu}_k = \int \mathbf{h}(\mathbf{x}_k) \, \mathsf{N}(\mathbf{x}_k \,|\, \mathbf{m}_k^-, \mathbf{P}_k^-) \, d\mathbf{x}_k$$

$$\mathbf{S}_k = \int (\mathbf{h}(\mathbf{x}_k) - \boldsymbol{\mu}_k)(\mathbf{h}(\mathbf{x}_k) - \boldsymbol{\mu}_k)^\mathsf{T} \, \mathsf{N}(\mathbf{x}_k \,|\, \mathbf{m}_k^-, \mathbf{P}_k^-) \, d\mathbf{x}_k + \mathbf{R}_k$$

$$\mathbf{C}_k = \int (\mathbf{x}_k - \mathbf{m}_k^-)(\mathbf{h}(\mathbf{x}_k) - \boldsymbol{\mu}_k)^\mathsf{T} \, \mathsf{N}(\mathbf{x}_k \,|\, \mathbf{m}_k^-, \mathbf{P}_k^-) \, d\mathbf{x}_k.$$

2. Then compute the following:

$$\mathbf{K}_k = \mathbf{C}_k \, \mathbf{S}_k^{-1}$$
$$\mathbf{m}_k = \mathbf{m}_k^- + \mathbf{K}_k \, (\mathbf{y}_k - \boldsymbol{\mu}_k)$$
$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{K}_k \, \mathbf{S}_k \, \mathbf{K}_k^\mathsf{T}.$$

- Special case of assumed density filtering (ADF).
- Multidimensional Gauss-Hermite quadrature $\Rightarrow$ Gauss Hermite Kalman filter (GHKF).
- Cubature integration $\Rightarrow$ Cubature Kalman filter (CKF).
- Monte Carlo integration $\Rightarrow$ Monte Carlo Kalman filter (MCKF).
- Gaussian process / Bayes-Hermite Kalman filter: Form Gaussian process regression model from set of sample points and integrate the approximation.
- Linearization, unscented transform, central differences, divided differences can be considered as special cases.

Gauss-Hermite Kalman Filter (GHKF) [1/2]

- One-dimensional Gauss-Hermite quadrature of order p:

$$\int_{-\infty}^{\infty} g(x)\, N(x \mid 0, 1)\, dx \approx \sum_{i=1}^{p} W^{(i)} g(x^{(i)}),$$

- $\xi^{(i)}$ are roots of pth order Hermite polynomial:

$$\begin{aligned}
H_0(x) &= 1 \\
H_1(x) &= x \\
H_2(x) &= x^2 - 1 \\
H_3(x) &= x^3 - 3x \ldots
\end{aligned}$$

- The weights are $W^{(i)} = p! / (p^2\, [H_{p-1}(\xi^{(i)})]^2)$.
- Exact for polynomials up to order $2p - 1$.

- Multidimensional integrals can be approximated as:

$$\int \mathbf{g}(\mathbf{x}) \, N(\mathbf{x} \,|\, \mathbf{m}, \mathbf{P}) \, d\mathbf{x}$$

$$= \int \mathbf{g}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}) \, N(\boldsymbol{\xi} \,|\, \mathbf{0}, \mathbf{I}) \, d\boldsymbol{\xi}$$

$$= \int \cdots \int \mathbf{g}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}) \, N(\xi_1 \,|\, 0, 1) \, d\xi_1 \times \cdots \times N(\xi_n \,|\, 0, 1) \, d\xi_n$$

$$\approx \sum_{i_1, \ldots, i_n} W^{(i_1)} \times \cdots \times W^{(i_n)} \mathbf{g}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}^{(i_1, \ldots, i_n)}).$$

- Needs p^n evaluation points.
- Gauss-Hermite Kalman filter (GHKF) uses this for evaluation of the Gaussian integrals.

- Postulate symmetric integration rule:

$$\int \mathbf{g}(\boldsymbol{\xi}) \; \mathrm{N}(\boldsymbol{\xi} \,|\, \mathbf{0}, \mathbf{I}) \, d\boldsymbol{\xi} \approx W \sum_i \mathbf{g}(c \, \mathbf{u}^{(i)}),$$

- The points $\mathbf{u}^{(i)}$ belong to the symmetric set [**1**] with generator $(1, 0, \dots, 0)$:

$$[\mathbf{1}] = \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \dots \begin{pmatrix} -1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \dots \right\}$$

- W is a weight and c is a parameter yet to be determined.

- Due to symmetry, all odd orders are integrated exactly.
- We only need to match the following moments:

$$\int N(\boldsymbol{\xi} \mid \mathbf{0}, \mathbf{I}) \, d\boldsymbol{\xi} = 1$$

$$\int \xi_j^2 \, N(\boldsymbol{\xi} \mid \mathbf{0}, \mathbf{I}) \, d\boldsymbol{\xi} = 1$$

- Thus we get the equations

$$W \sum_i 1 = W \, 2n = 1$$

$$W \sum_i [c \, u_j^{(i)}]^2 = W \, 2c^2 = 1$$

- Hence the following rule is exact up to third degree:

$$\int \mathbf{g}(\boldsymbol{\xi}) \, N(\boldsymbol{\xi} \mid \mathbf{0}, \mathbf{I}) \, d\boldsymbol{\xi} \approx \frac{1}{2n} \sum_i \mathbf{g}(\sqrt{n} \, \mathbf{u}^{(i)}).$$

- The resulting Gaussian integral rule:

$$
\int \mathbf{g}(\mathbf{x})\, \mathsf{N}(\mathbf{x}\,|\,\mathbf{m},\mathbf{P})\, d\mathbf{x}
$$
$$
= \int \mathbf{g}(\mathbf{m}+\sqrt{\mathbf{P}}\,\xi)\, \mathsf{N}(\xi\,|\,\mathbf{0},\mathbf{I})\, d\xi
$$
$$
\approx \frac{1}{2n}\sum_{i=1}^{2n}\mathbf{g}(\mathbf{m}+\sqrt{\mathbf{P}}\,\xi^{(i)}),
$$

where

$$
\xi^{(i)} = \left\{ \begin{array}{ll} \sqrt{n}\,\mathbf{e}_i & ,\quad i=1,\ldots,n \\ -\sqrt{n}\,\mathbf{e}_{i-n} & ,\quad i=n+1,\ldots,2n, \end{array} \right.
$$

where $\mathbf{e}_i$ denotes a unit vector to the direction of coordinate axis i.

Cubature Kalman filter: Prediction step

1. Form the sigma points as:

$$\mathcal{X}_{k-1}^{(i)} = \mathbf{m}_{k-1} + \sqrt{\mathbf{P}_{k-1}}\,\boldsymbol{\xi}^{(i)} \qquad i = 1, \ldots, 2n.$$

2. Propagate the sigma points through the dynamic model:

$$\hat{\mathcal{X}}_k^{(i)} = \mathbf{f}(\mathcal{X}_{k-1}^{(i)}). \quad i = 1 \ldots 2n.$$

3. Compute the predicted mean and covariance:

$$\mathbf{m}_k^- = \frac{1}{2n} \sum_{i=1}^{2n} \hat{\mathcal{X}}_k^{(i)}$$

$$\mathbf{P}_k^- = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\mathcal{X}}_k^{(i)} - \mathbf{m}_k^-)\,(\hat{\mathcal{X}}_k^{(i)} - \mathbf{m}_k^-)^\mathsf{T} + \mathbf{Q}_{k-1}.$$

Cubature Kalman filter: Update step

1. Form the sigma points:

$$\mathcal{X}_k^{-(i)} = \mathbf{m}_k^- + \sqrt{\mathbf{P}_k^-} \, \boldsymbol{\xi}^{(i)}, \qquad i = 1, \ldots, 2n.$$

2. Propagate sigma points through the measurement model:

$$\hat{\mathcal{Y}}_k^{(i)} = \mathbf{h}(\mathcal{X}_k^{-(i)}), \quad i = 1 \ldots 2n.$$

Cubature Kalman filter: Update step (cont.)

③ Compute the following:

$$\boldsymbol{\mu}_k = \frac{1}{2n} \sum_{i=1}^{2n} \hat{\mathcal{Y}}_k^{(i)}$$

$$\mathbf{S}_k = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)(\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)^\mathsf{T} + \mathbf{R}_k$$

$$\mathbf{C}_k = \frac{1}{2n} \sum_{i=1}^{2n} (\mathcal{X}_k^{-(i)} - \mathbf{m}_k^-)(\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)^\mathsf{T}$$

$$\mathbf{K}_k = \mathbf{C}_k \mathbf{S}_k^{-1}$$

$$\mathbf{m}_k = \mathbf{m}_k^- + \mathbf{K}_k \left[\mathbf{y}_k - \boldsymbol{\mu}_k \right]$$

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

- Cubature Kalman filter (CKF) is a special case of UKF with $\alpha = 1$, $\beta = 0$, and $\kappa = 0$ – the mean weight becomes zero with these choices.
- Rule is exact for third order polynomials (multinomials) – note that third order Gauss-Hermite is exact for fifth order polynomials.
- UKF was also originally derived using similar way, but is a bit more general.
- Very easy algorithm to implement – you can even recall the rule by heart.

Summary

- **Unscented transform** (UT) approximates transformations of Gaussian variables by propagating sigma points through the non-linearity.

- In UT the mean and covariance are approximated as linear combination of the sigma points.

- The unscented Kalman filter uses unscented transform for computing the approximate means and covariance in non-linear filtering problems.

- A non-linear transformation can also be approximated with Gaussian moment matching.

- Gaussian filter is based on matching the moments with numerical integration $\Rightarrow$ many kinds of Kalman filters.

- Gauss-Hermite Kalman filter (GHKF) and Cubature Kalman filter (CKF) are examples of them.

Unscented/Cubature Kalman Filter (UKF/CKF): Example

- Recall the discretized pendulum model

$$\begin{pmatrix} x_k^1 \\ x_k^2 \end{pmatrix} = \underbrace{\begin{pmatrix} x_{k-1}^1 + x_{k-1}^2 \, \Delta t \\ x_{k-1}^2 - g \, \sin(x_{k-1}^1) \, \Delta t \end{pmatrix}}_{\mathbf{f(x_{k-1})}} + \begin{pmatrix} 0 \\ q_{k-1} \end{pmatrix}$$

$$y_k = \underbrace{\sin(x_k^1)}_{\mathbf{h(x_k)}} + r_k,$$

- $\Rightarrow$ *Matlab demonstration*