TP1 Kalman Filtering

UV SDATA - 2021 - C. Garnier - CERI SN

Before you start

- Read well the slides of the course.
- The objective of TP1 is to perform sequential Bayesian estimation in case of a linear Gaussian state-space model using Kalman filtering.
- Be curious : Play with the parameters: state noise variance, observation noise variance, initial values....

1. Target tracking using Kalman filtering

The goal is to track a target in a plane.

- A sensor regularly provides the location of the target on a plane along two axes with a period $T_e = 1$: y_k^1 along the axis 1 and y_k^2 along the axis 2. But the sensor is not perfect and the provided locations are noisy.
- We assume that the trajectory of the target at time $kT_e = k$ is determined by the evolution of the following quantities: the position x_k^1 and the velocity v_k^1 along the axis 1, the position x_k^2 and the velocity v_k^2 along the axis 2.

1.1. Target tracking model

The tracking problem is assumed linear and Gaussian, and is modeled by the following equations:

State equation: The hidden state $\mathbf{x}_k \in \mathbb{R}^4$ is the vector: $\mathbf{x}_k = \begin{bmatrix} x_k^1 & v_k^1 & x_k^2 & v_k^2 \end{bmatrix}^T$ which evolves according to the following state equation:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{u}_k \tag{1}$$

where

$$\mathbf{F} = \begin{bmatrix} 1 & T_e & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_e \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Interpretation: The position of the target at time k is equal to its position at previous time $k-1$ increased by the distance travelled at the current velocity, plus a noise term. The target velocity is quasi constant.

The state noise $\mathbf{u}_k \in \mathbb{R}^4$ is Gaussian: $\mathbf{u}_k \sim \mathcal{N}(\mathbf{0}_{4\times 1}, \mathbf{\Sigma_u})$.

In tracking problems, we usually choose: $\mathbf{\Sigma_u} = \sigma^2 \begin{bmatrix} \frac{T_e^3}{3} & \frac{T_e^2}{2} & 0 & 0 \\ \frac{T_e^2}{2} & T_e & 0 & 0 \\ 0 & 0 & \frac{T_e^3}{3} & \frac{T_e^2}{2} \\ 0 & 0 & \frac{T_e^2}{2} & T_e \end{bmatrix} = \sigma^2 \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{bmatrix}$

Observation equation: The observation (or measurement) $\mathbf{y}_k \in \mathbb{R}^2$ is the vector: $\mathbf{y}_k = \begin{bmatrix} y_k^1 & y_k^2 \end{bmatrix}^T$ which is related to the hidden state by the following measurement equation:

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k \tag{2}$$

where

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Interpretation: The observation provided by the sensor is the position of the target at time k plus a noise term, which models the measure imperfection.

The observation noise $\mathbf{w}_k \in \mathbb{R}^2$ is Gaussian: $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}_{2\times 1}, \mathbf{\Sigma_w})$.

In tracking problems, we usually choose: $\mathbf{\Sigma_w} = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$

1.2. Interpretation of the state space model

Derive the expressions of x_k^1, v_k^1, x_k^2 and v_k^2 using Eq. (1) and check the interpretation of the state equation.

Derive the expressions of y_k^1 and y_k^2 using Eq. (2) and check the interpretation of the observation equation.

1.3. Initialization of the model parameters

Create and initialize the following parameters: $T_e = 1$, $\sigma = 1$, $\sigma_1 = 30$, $\sigma_2 = 30$, $T = 100$ (number of time steps, i.e. duration of the simulation).

Create and initialize the following matrices: $\mathbf{F}$, $\mathbf{\Sigma_u}$, $\mathbf{H}$, $\mathbf{\Sigma_w}$ as given previously.

1.4. Generation of a trajectory of the target

Initialize the hidden state by $\mathbf{x}_0 = \begin{bmatrix} 3 & 40 & -4 & 20 \end{bmatrix}^T$.

Generate the hidden states $\mathbf{x}_k = \begin{bmatrix} x_k^1 & v_k^1 & x_k^2 & v_k^2 \end{bmatrix}^T$ for $k = 1 \ldots T$ using Eq. (1) and the model parameters defined previously.

Store all the hidden states in a matrix of size $4 \times T$. The evolution of the position in the plane (x_k^1, x_k^2) represents the true trajectory of the target.

Plot the true target trajectory in the plane.

1.5. Generation of the observations provided by the sensor

Generate the observations $\mathbf{y}_k = \begin{bmatrix} y_k^1 & y_k^2 \end{bmatrix}^T$ provided by the sensor for $k = 1 \ldots T$ using Eq. (2) and the model parameters previously defined.

Store all the observations in a matrix of size $2 \times T$. These observations give a noisy representation of the target trajectory.

On the same figure, plot the true trajectory and the observed noisy trajectory (with different colors).

1.6. Implementation of the Kalman filter

Now the goal is to estimate the trajectory of the target from the noisy observations provided by the sensor using the Kalman Filter.

The tracking state-space model is linear and Gaussian, so the filtering distribution is Gaussian: $p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$. Write the code corresponding the Kalman filter which recursively estimates the filtering distribution parameters: $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$.

First the program must initialize the mean and the covariance matrix. We choose: $\hat{\mathbf{x}}_{0|0} = \mathbf{x}_0$ and $\mathbf{P}_{0|0} = \mathbf{I}_{4\times 4}$.

Then at each time step k, the program must perform the prediction and update steps as follows:

1. **Prediction** step:
   Calculate the parameters of the Gaussian predictive distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) = \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$:
   - Predictive mean: $\mathbf{x}_{pred} = \hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{x}}_{k-1|k-1}$
   - Predictive covariance matrix: $\mathbf{P}_{pred} = \mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{\Sigma_u}$

   The mean is propagated by the transition matrix $\mathbf{F}_k$ and the **uncertainty** (variance) is also propagated by $\mathbf{F}_k$ and increased by the covariance $\Sigma_\mathbf{u}$ of the state noise.
2. **Update** step:
   Calculate the parameters of the Gaussian filtering distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$:
   - Updated/estimated mean: $\mathbf{x}_{est} = \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \left(\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}\right)$
   - Updated/estimated covariance matrix: $\mathbf{P}_{est} = \mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\, \mathbf{P}_{k|k-1}$

   where $\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{\Sigma_w}\right)^{-1}$ is the Kalman gain.
   The predictive mean is corrected in the direction of the residual error $\left(\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}\right)$ and the predictive covariance matrix is adjusted using the Kalman gain.

Store the estimated states $\mathbf{x}_{est} = \hat{\mathbf{x}}_{k|k}$ for all the time steps $k = 1 \ldots T$ in a matrix of size $4 \times T$.

1.7. Performance analysis

On the same figure, plot the true trajectory, the observed noisy trajectory and the estimated trajectory (with different colors) in the plane.

On different figures, plot the true values and the estimated values of respectively x_k^1, v_k^1, x_k^2 and v_k^2 versus the time steps.

Compute the root mean squared error (RMSE) of the Kalman filter estimation at each time step:

$$RMSE_k = \|\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}\| = \sqrt{(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})^T (\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})}$$

and then compute the average RMSE: $\text{RMSE} = \frac{1}{T} \sum_{k=1}^{T} \|\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}\|$.

Run the Kalman filter for different parameter and initialization values and analyse how these values affect the performance. In particular, vary the state noise variance σ^2, the observation noise variances σ_1^2 and σ_2^2 and the initial mean $\hat{\mathbf{x}}_{0|0}$ in the Kalman filter.