Kalman Filter

Dhruv Patel

May 10, 2020

What is Kalman Filter ?

Kalman Filter or simply *KF*, is one of the many techniques to implement Bayes Filters. It was invented by Swerling (1958) and Kalman (1960) as a technique for filtering and prediction in *linear Gaussian Systems*. The Kalman Filter implements belief computation for continuous states. It is not applicable to discrete or hybrid state spaces.

```
Algorithm Kalman Filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):
    # Prediction Step : ref. as Motion Model
    $\overline{\mu}_t = A_t * \mu_{t-1} + B_t * u_t$
    $\overline{\Sigma}_t = A_t * \Sigma_{t-1} * A_t^T + R_t$
    # Update Step : ref. as Measurement Model
    $K_t = \overline{\Sigma}_t * C_t^T * (C_t * \Sigma_t * C_t^T + Q_t)^{-1}$
    $\mu_t = \overline{\mu}_t + K_t * (z_t - C_t * \overline{\mu}_t)$
    $\Sigma_t = (I - K_t * C_t) * \overline{\Sigma}_t$
    return $\mu_t, \Sigma_t$
```

The Kalman Filter represents beliefs by the moments parameterization : At time t, the belief $bel(x_t)$is represented by the mean μ_t and the covariance Σ_t. The input of the Kalman filter is the belief at time $t-1$, represented by μ_{t-1} and Σ_{t-1}. To update these parameters, Kalman filters require the control u_t and the measurement z_t. The output is the belief at time t, represented by μ_t and Σ_t. This predicted belief μ_t and Σ_t is calculated representing the belief $\overline{bel}(x_t)$ one time step later, but before incorporating the measurement z_t. This belief is obtained by incorporating the control u_t. The update of the covariance considers the fact that states depend on previous states through the linear matrix A_t. This matrix is multiplied twice into the covariance, since the covariance is a quadratic matrix.

The belief $\overline{bel}(x_t)$ is subsequently transformed into the desired belief $bel(x_t)$, by incorporating the measurement z_t. The variable K_t, is called *Kalman Gain*. It specifies the degree to which the measurement is incorporated into the new state estimate. The key concept here is the *innovation*, which is the difference between the actual measurement z_t and the expected measurement C_t. Finally, the new covariance of the posterior belief is calculated, adjusting for the information gain resulting from the measurement.

The *Kalman Filter* is computationally quite efficient. In many applications - such as the robot mapping applications - the measurement space is much lower dimensional than the state space, and the update is dominated by the $O(n^2)$ operations.

A brief explanation about the variables

The state transition probability $p(x_t|u_t, x_{t-1})$ must be a linear function in its arguments with added Gaussian noise. This is expressed by the following equation :

$$x_t = A_t * x_{t-1} + B_t * u_t + \varepsilon_t \tag{1}$$

Here, x_t and x_{t-1} are state vectors, and u_t is the control vector at time t. These vectors are column vectors. They are of the form

$$x_t = \begin{pmatrix} x_{1,t} \\ x_{2,t} \\ . \\ . \\ x_{n,t} \end{pmatrix} \text{ and } u_t = \begin{pmatrix} u_{1,t} \\ u_{t,2} \\ . \\ . \\ u_{m,t} \end{pmatrix} \tag{2}$$

A_t and B_t are matrices. A_t is a square matrix of size $n \times n$, where n is the dimension of the state vector x_t. B_t is of size $n \times m$, with m being the dimension of the control vector u_t. By multiplying the state and control vector with the matrices A_t and B_t, respectively, the state transition function becomes *linear* in its arguments. Thus, Kalman Filters assume linear system dynamics.

The random variable ε_t in (1) is a Gaussian random vector that models the uncertainty introduced by the state transition. It is of the same dimension as the state vector. It has zero mean, and its covariance will be denoted by R_t. A state transition probability of the form (1) is called a *linear Gaussian*, to reflect the fact that it is linear in its arguments with additive Gaussian noise.

The measurement probability $p(z_t|x_t)$ must also be *linear* in its arguments, with added Gaussian noise :

$$z_t = C_t * x_t + \delta_t \tag{3}$$

Here, C_t is a matrix of size $k \times n$, where k is the dimension of the measurement vector z_t. The vector δ_t describes the measurement noise. The distribution of δ_t is a multivariate Gaussian with zero mean and covariance Q_t.

Given Problem

Here, we have been given a simple target tracking problem in one-dimensional space. The state contains three components : position (one-dimensional), velocity and acceleration and it can be expressed as below.

$$x = \begin{pmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{pmatrix} \tag{4}$$

where, x_k is the position, $\dot{x}_k$ is the velocity and $\ddot{x}_k$ is the acceleration at time 'k'. T is the size of sample time step.

The process equation for target motion is given by the following kinematic equation:

$$\begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x_{k-1} \\ \dot{x}k-1 \\ \ddot{x}k-1 \end{bmatrix} + \upsilon_{k-1} \tag{5}$$

and

$$\upsilon_{k-1} \sim N(0,Q) \tag{6}$$

where, $Q = \sigma^2 * \begin{bmatrix} \frac{T^4}{4} & \frac{T^3}{2} & \frac{T^2}{2} \\ \frac{T^3}{2} & 2T^3 & T^2 \\ \frac{T^2}{2} & T^2 & T^2 \end{bmatrix}$, σ represents intensity of Gaussian noise.

Targets are usually tracked with the help of sensors such as radars or lidars which provide the position of the target. Hence, the measurement equation can be written as:

$$y_k = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} + \omega \tag{7}$$

where, y_k is the measurement and $\omega \sim N(0,R)$ is the measurement noise.

For the given problem, obtain the estimate of its state over a period of $20\ sec$. Assume time step $T = 0.1\ sec$.