Kalman Filtering

Motivation

KALMAN FILTERING

Motivation

The Bayes Filter calculates the belief across the *entire* state space.

For practical navigation purposes, while calculable, the problem is intractable.

- Three position states, velocities, angular velocities…

Recall some classical statistics:

- Because we model the navigation state as a collection of *independent* random variables, the central limit theorem states that if we measure our state enough times the samples will resemble a normal distribution
- When modeling something probabilistically, a normal distribution is a reasonable model.
- Alternatively, we can make the design choice to *parameterize* our state space as an estimate (mean) and uncertainty (covariance).

Assumptions

With this motivation to parameterize as a normal distribution and simply the state space we can (must) make two assumptions:

1. The prior state is represented as a normal distribution
   - $p(x_0) \sim N(\mu_o, \Sigma_0)$
2. The process model and measurement model are *linear* with additive *normal* white noise
   - $x_t = A_t x_{t-1} + B_t u_t + n_t$
   - $z_t = C_t x_t + v_t$

Gaussian Random Variables

A multivariate normal distribution takes the form where x is a vector of random variables:

$$f_x(x) = \frac{1}{(2\pi)^{\frac{n}{2}}\sqrt{\det(\Sigma)}} e^{-\frac{(x-\mu)^T\Sigma^{-1}(x-\mu)}{2}}$$

$f_x(x)$ is a function of x, μ, and Σ, thus we can say it is fully parameterized by μ and Σ

Affine (linear) transformations of Gaussian (normal) distributions remain Gaussian

- If $x \sim \mathcal{N}(\mu_x, \Sigma_x)$ and $y = Ax + b$
- Then y$\sim\mathcal{N}(\mu_y, \Sigma_y)$, $\mu_y = A\mu_x$ and $\Sigma_y = A\Sigma_x A^T$

Affine transformation example

Expectation is a linear operator:

- $x \sim \mathcal{N}(\mu_x, \Sigma_x)$
- $E[X] = \int p(x)xdx = \mu_x$

$y = Ax + b$

- $\mu_y = E[y] = E[Ax + b] = AE[x] + b = A\mu_x + b$
- $\Sigma_y = E\left[(y - \mu_y)(y - \mu_y)^T\right] = E[(Ax + b \; - A\mu_x - b)(Ax + b - A\mu_x - b)^T]$
  $= AE[(x - \mu_x)(x - \mu_x)^T]A^T = A\Sigma_x A^T$

Properties

Other properties and formations of transformations and probability distributions also hold (independence, summation)

The joint distribution of two normal random variables is not necessarily a normal distribution

Conditional distributions:

- $X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$
- Conditional probability density ($f(x_1 | X_2 = x_2)$) is a normal distribution parameterized as such:
  - $\mu_{X_1|X_2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2)$
  - $\Sigma_{X_1|X_2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$

Derivation

KALMAN FILTERING

Assumptions and summary

Prior state is a normal distribution: $p(x_0) \sim \mathcal{N}(\mu_0, \Sigma_0)$

The process model $f(x_t | x_{t-1}, u_t)$ is linear with additive normal white noise:

- $x_t = A_t x_{t-1} + B_t u_t + \mathcal{N}(0, Q_t)$

The measurement model $g(z_t | x_t)$ is linear with normal white noise:

- $z_t = C_t x_t + \mathcal{N}(0, R_t)$

Implications:

- Linear transformation of normal distributions remain normal
- Propagation is a straightforward linear transformation and preserves the normal distribution
- If we can find a way to relate the measurements to the state in a linear fashion, we can use this as a real time estimator and tractable implementation of a Bayesian Filter.

Prediction step

Recall Bayes: $p(x_t|z_{1:t-1}, u_{1:t}) = \int f(x_t|x_{t-1}, u_t)p(x_{t-1}|z_{1:t-1}, u_{1:t-1})dx_{t-1}$

- $p(x_{t-1}|z_{1:t-1}, u_{1:t-1}) \sim \mathcal{N}(\mu_{t-1}, \Sigma_{\mathrm{t}-1})$
- $f(x_t|x_{t-1}, u_t) = A_t x_{t-1} + B_t u_t + \mathcal{N}(0, Q_t)$

Parameterize and apply expected value and covariance operations:

- $\overline{\mu_t} = A\mu_{t-1} + Bu_t$
- $\overline{\Sigma_t} = A\Sigma_{t-1}A^T + Q$

Update Step

Recall Bayes: $p(x_t|z_{1:t}, u_{1:t}) = \frac{g(z_t|x_t)p(x_t|z_{1:t-1}, u_{1:t})}{\int g\left(z_t\middle|x_t'\right)p\left(x_t'\middle|z_{1:t-1}, u_{1:t}\right)dx_t'}$

- $g(z_t|x_t) = z_t = C_t\bar{x}_t + \mathcal{N}(0, R)$

If we don't have a measurement, the best update is to just propagate ($x_t = \bar{x}_t$)

- $\begin{bmatrix} x_t \\ z_t \end{bmatrix} = \begin{bmatrix} I & 0 \\ C & I \end{bmatrix} \begin{bmatrix} \bar{x}_t \\ \mathcal{N}(0, R) \end{bmatrix}$

Is this still a normal distribution?

- $\mu = \begin{bmatrix} \bar{\mu}_t \\ C\bar{\mu}_t \end{bmatrix}$
- $\Sigma = \begin{bmatrix} I & 0 \\ C & I \end{bmatrix} \begin{bmatrix} \bar{\Sigma}_t & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} I & C^T \\ 0 & I \end{bmatrix} = \begin{bmatrix} \bar{\Sigma}_t & \bar{\Sigma}_t C^T \\ C\bar{\Sigma}_t & C\bar{\Sigma}_t C^T + R \end{bmatrix}$

Update Step

Yes! The update of x_t conditioned on z_t is normal:

- $\mu_{x_t|z_t} = \bar{\mu}_t + \bar{\Sigma}_t C^T (C \bar{\Sigma}_t C^T + R)^{-1} (z_t - C \bar{\mu}_t)$
- $\Sigma_{x_t|z_t} = \bar{\Sigma}_t - \bar{\Sigma}_t C^T (C \bar{\Sigma}_t C^T + R)^{-1} C \bar{\Sigma}_t$

For notation's sake we'll define the matrix K as the Kalman gain:

- $K = \bar{\Sigma}_t C^T (C \bar{\Sigma}_t C^T + R)^{-1}$

Update step:

- $K_t = \bar{\Sigma}_t C^T (C \bar{\Sigma}_t C^T + R)^{-1}$
- $\mu_t = \bar{\mu}_t + K_t (z_t - C \bar{\mu}_t)$
- $\Sigma_t = \bar{\Sigma}_t - K_t C \bar{\Sigma}_t$

Kalman Gain

$K = \bar{\Sigma}_t C^T (C\bar{\Sigma}_t C^T + R)^{-1}$

In plain English, a matrix of weightings for how much to trust the sensor against the prediction.

For a perfect sensor: $R = 0$

- $K_t = \bar{\Sigma}_t C^T (C\bar{\Sigma}_t C^T + R)^{-1} = C^{-1}$
- $\mu_t = \bar{\mu}_t + K_t(z_t - C\bar{\mu}_t) = C^{-1} z_t$
- $\Sigma_t = \bar{\Sigma}_t - K_t C \bar{\Sigma}_t = \bar{\Sigma}_t - C^{-1} C \bar{\Sigma}_t = 0$

For no (bad) sensor: $R \to \infty$

- $K_t = \bar{\Sigma}_t C^T (C\bar{\Sigma}_t C^T + R)^{-1} \to 0$
- $\mu_t = \bar{\mu}_t + K_t(z_t - C\bar{\mu}_t) \to \bar{\mu}_t$
- $\Sigma_t = \bar{\Sigma}_t - K_t C \bar{\Sigma}_t = \bar{\Sigma}_t - 0\bar{\Sigma}_t = \bar{\Sigma}_t$

Some facts

If the distribution is not Gaussian, the Kalman filter is the minimum variance linear estimator

The variance never increases due to receiving a measurement

The variance update is independent of the measurement realization

Continuous Time Systems

KALMAN FILTERING

Discrete vs. Continuous Time

DISCRETE

Events occur at distinct points in time

Typically have evenly spaced time intervals

Represented by a *difference* equation:

- $f_t = af_{t-1} + bf_{t-2}$

CONTINUOUS

Events may occur at infinitesimally small time intervals (aka continuously).

Represented by a *differential* equation

- $\ddot{x} = a\dot{x} + bx$

Issue with Kalman Filter

Problem

- Kalman Filter is a digital filter designed to work in a given cycle (namely on a computer)
- Kalman Filter is thus *discrete*
- Real world motion exists in continuous time

Solution

- Convert the continuous time system into a discrete system
- Reformulate the Kalman Filter to work with continuous dynamics

Continuous to Discrete

Fortunately, the first-order Markov assumption makes the Kalman Filter system model a first order differential equation

- $\dot{x} = f\big(x, u, \mathcal{N}(0, Q)\big) = Ax + Bu + E\mathcal{N}(0, Q)$

Discretize using one step Euler integration method

- In continuous time: $x(t) = \int \dot{x}dt = x_0 + \dot{x}\delta t$
- In discrete time: $x_t = x_{t-1} + \dot{x}\delta t = x_{t-1} + (Ax_{t-1} + Bu_t + E\eta_t)\delta t$
- Collecting terms: $x_t = (I + \delta t A)x_{t-1} + \delta t B u_t + \delta t E \eta_t = F x_{t-1} + G u_{t-1} + V\eta_t$

Now in the Kalman Filter prediction step:

- $\bar{\mu}_t = F\mu_{t-1} + Gu_t$
- $\bar{\Sigma}_t = F\Sigma_{t-1}F^T + VQV^T$

Continuous time example

Second order, one dimensional system with state vector $\mathbf{x} = [s, \dot{s}]^T$ and control input $\mathbf{u} = \dot{s}$

- $\dot{\boldsymbol{x}} = f(\mathbf{x}, \mathbf{u}, \eta) = A\mathbf{x} + B\mathbf{u} + E\eta$

  $$= \begin{bmatrix} \dot{s} \\ \ddot{s} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + E\eta$$

- $F = (I + \delta t\ A) = \begin{bmatrix} 1 & \delta t \\ 0 & 1 \end{bmatrix}$
- $G = \delta t B = \delta t \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \delta t \end{bmatrix}$
- $V = \delta t\ E$

Kalman Filter prediction equations

- $\bar{\mu}_t = F\mu_{t-1} + Gu_t$
- $\bar{\Sigma}_t = F\Sigma_{t-1}F^T + VQV^T$

The notation makes it look complicated.

- Often a continuous time system is just automatically converted straight to discrete, but some authors still use A to indicate the state transition matrix
- One-step Euler integration is as simple as it appears

An example

2D PLANER MOTION, 2 DEGREES OF FREEDOM

An example

Simple case: 2D planer motion with two degrees of freedom, no control input, acceleration modeled as noise

- $\boldsymbol{x} = (p_x, p_y, v_x, v_y)$
- $\boldsymbol{z} = (z_x, z_y)$

Continuous time system is clearly linear:

- $\dot{x}_t = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}_t x_t + \eta_t$
- $F = (I + A\delta t) = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_t$

This follows our intuition of basic kinematics:

- $\bar{x} = x + vt$

No control inputs modeled, so B matrix is zero.

Acceleration is modeled as noise so E matrix is one on the diagonals for the velocity terms:

- $E = \begin{bmatrix} \mathbf{0}_2 & \mathbf{0}_2 \\ \mathbf{0}_2 & I_2 \end{bmatrix}$
- $V = E\delta t = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \delta t & 0 \\ 0 & 0 & 0 & \delta t \end{bmatrix}$

Measurement matrix is straightforward:

- $C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$

An example

Kalman Filter prediction equations:

$$\bar{\mu}_t = \overline{\begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix}}_t = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_t \begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix}_{t-1}$$

$$\bar{\Sigma}_t = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_t \Sigma_t \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_t^T + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \delta t & 0 \\ 0 & 0 & 0 & \delta t \end{bmatrix} Q \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \delta t & 0 \\ 0 & 0 & 0 & \delta t \end{bmatrix}^T$$

An example

Kalman Filter update equations

$$K = \bar{\Sigma}_t \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}^T \left(\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \bar{\Sigma}_t \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}^T + R \right)^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t \left(z_t - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \bar{\mu}_t \right)$$

$$\Sigma_t = \bar{\Sigma}_t - K_t \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \bar{\Sigma}_t$$

This is a basic example of two-dimensional target tracking

- Can easily be extend to three dimensions
- Alternatively, can be used for navigation when you have an external reference for position and orientation

Summary

Summary

The Bayes Filter is largely intractable and impractical for even fairly simple and lower dimensional systems

Gaussian-based filters (namely the Kalman Filter) represent the first mathematically tractable implementation of a Bayesian Filter.

Kalman Filters assume three things:

- The prior state is best represented by a normal distribution parameterized by a mean and covariance
- The process model and measurement model are linear with additive white (normally distributed) noise

Summary

ADVANTAGES

Kalman Filters are simple and intuitive.

Purely matrix operations and thus computationally efficient, even for higher-dimensional systems.

DISADVANTAGES

Assumes everything is linear and normally distributed

Unimodal distribution, unlike Bayes, and cannot process multiple hypothesis