Robot Autonomy Lecture - Kalman Filters
March 22, 2017

Amit Agarwal (amita1)
Cyrus Liu (xiyuanl1)
Kazuya Otani (kotani)

Kalman Filters can be used for something as trivial as correctly sensing the temperature of a room to something as complex as measuring the trajectory in space travel.

1. Why do we need Kalman filters?

- Motion planning assumes exact state information
- Sensors carry only partial information, often noisy
- Initial estimates of robot state are uncertain

We need a way to recover state information from sensor data!

Kalman filters can:

- Fuse information from multiple sensors
- Incorporate indirect measurements of state information
- Deal with uncertainty

Use y (sensor measurements) and u (control inputs) to estimate x (states).

We may also not be completely certain about initial state (prior = Gaussian)

Key idea

- Represent initial estimate as probability distribution
- Propagate state estimate forward using known dynamics equation
- Use sensor feedback to update probability distribution

To do this, Kalman filters use Bayes' rule

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

2. Recall: Gaussians

One-dimensional Gaussian distribution

$$f(x) = ae^{-\frac{(x-b)^2}{2c^2}}$$

Multi-dimensional Gaussian distribution

$$f_{\mathbf{x}}(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}}$$

$$= \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{|2\pi\boldsymbol{\Sigma}|}},$$

Good thing about Gaussians: product of two Gaussians is also Gaussian! This means that if the prior probability distribution is Gaussian, the noise is Gaussian and the calculations are linear, then the posterior distribution will be Gaussian too. This makes the math simple.

3. Kalman Filters

Kalman filters are “best linear unbiased estimators”. This means that if the dynamics and measurement functions are linear, and the additive noise is unbiased (assumed to be Gaussian), Kalman filters minimize least square error

Notation

- A - Transition function (Dynamic)
- B - Motion model
- H - Measurement function (Sensor model)
- $\epsilon = N(0, R)$ - Transition/process noise. R is the variance
- $\delta = N(0, Q)$ - Measurement noise. Q is the variance

Predict

$$\hat{x}_t = Ax_{t-1} + Bu_t$$

$$\hat{\Sigma}_t = A\Sigma_{t-1}A^T + R_t$$

Update

$$K_t = \hat{\Sigma}_t H^T (H\hat{\Sigma}_t H^T + Q_t)^{-1}$$

$$x_t = \hat{x}_t + K(y_t - H\hat{x}_t)$$

$$\Sigma_t = (I - K_t H)\hat{\Sigma}_t$$

Consider:

If Q=0 (very confident in measurements)

$$K_t = \hat{\Sigma}_t H^T (H_t \hat{\Sigma}_t H_t^T)^{-1} = I H_t^{-1} = H_t^{-1}$$
$$x_t = H^{-1} y_t$$
$$\Sigma_t = (I - K_t H)\hat{\Sigma}_t = 0$$

If Q=∞ (don't believe measurements at all)

$$K_t = \hat{\Sigma}_t H^T (0) = 0$$
$$x_t = \hat{x}_t$$
$$\Sigma_t = I\hat{\Sigma}_t$$

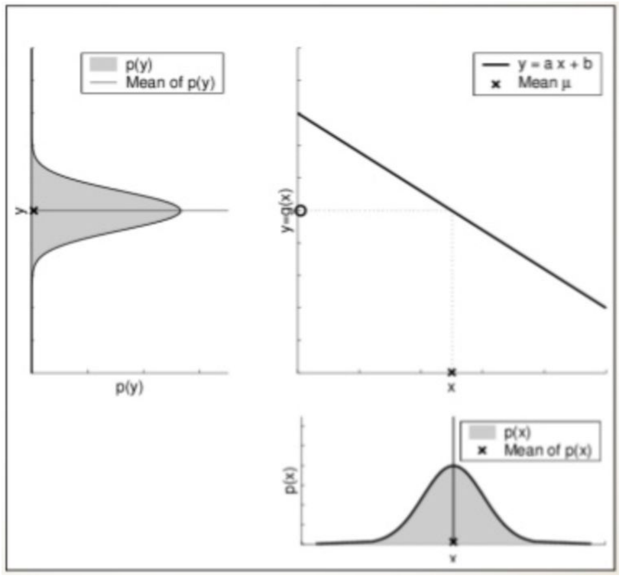
Key Assumptions

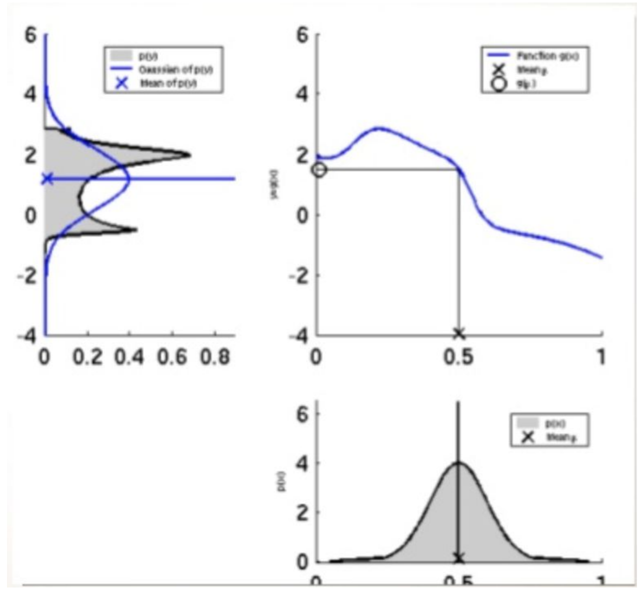
- Discrete-time linear system
- All noise is Gaussian

Is this always true? No, but in many cases this approximation is close enough.

4. Extended Kalman Filter

Problem: The world is not linear. When the system is highly nonlinear, using a Kalman filter may not lead to desired results (see figures below).





Idea: Do first order Taylor approximation (i.e. linearize!), then use Kalman filter

- Jacobian
- Basically, fit a Gaussian to a non-Gaussian distribution
- Would break when dynamics are really nonlinear, or if distribution is very non-Gaussian (ex. multi-modal)

Notation

$f(x, u)$ - dynamics function

$h(x)$ - measurement function

$$F_{t-1} = \frac{\partial_f}{\partial_x}\bigg|_{\hat{x}_{t-1}, u_{t-1}}$$

$$H_t = \frac{\partial_h}{\partial_x}\bigg|_{x_t}$$

Predict

$$\hat{x_t} = f(x_{t-1}, u_t)$$

$$\hat{\Sigma}_t = F_{t-1}\Sigma_{t-1}F_{t-1}^T + R_t$$

Update

$$K_t = \hat{\Sigma}_t H_t^T (H_t \hat{\Sigma}_t H_t^T + Q_t)^{-1}$$

$$x_t = \hat{x}_t + K(y_t - h(\hat{x}_t))$$
$$\Sigma_t = (I - K_t H_t)\hat{\Sigma}_t$$

5. Other kinds of state estimators

- Unscented Kalman Filter
    - Samples from a Gaussian distribution at each timestep, while allowing nonlinear transition and measurement functions. A bit more computationally expensive, but could be more accurate than KF or EKF if the dynamics and measurements function are highly nonlinear.
- Particle filters
    - Maintains a set of “particles” and propagates them with the dynamics model plus noise, weighting them by how well they agree with new measurements. This allows nonlinear transition and measurement functions, as well as multi-modal probability distributions (which Kalman filters do not, since Gaussians are defined by one mean and a variance).

References

- Video series illustrating Kalman filter in simple terms: http://studentdavestutorials.weebly.com/kalman-filter-with-matlab-code.html
- Wikipedia: https://en.wikipedia.org/wiki/Kalman_filter
- Cool application: https://www.youtube.com/watch?v=Qxu_y8ABajQ