Kalman Filter

(자율주행 핵심기술 SLAM 단기강좌)

한양대학교

남해운

(hnam@hanyang.ac.kr)

Bayes Filters

- The Bayes filter is not a practical algorithm
  - Gaussian Filters
    - Implementations of the Bayes Filter obtained by means of a Gaussian approximation of the pdf
      - Unimodal approximations completely defined through mean μ and covariance Σ
      - Ok for Position Tracking or in the global case if measurements allow to restrict the estimation to a small region (e.g. RFID)
    - Kalman Filter (KF)
    - Extended Kalman Filter (EKF)
    - Unscented Kalman Filter (UKF)
    - Extended Information Filter (EIF)
  - Non Parametric Filters
    - Histogram Filter (HF)
    - Particle Filter (PF)

Kalman Filter

- Linear Gaussian systems
  - The best studied techniques for implementing Bayes filter
  - The belief at time t is represented by mean μ_t and covariance Σ_t
- Posteriors are Gaussian if three properties hold
  - The state transition probability $p(x_t \mid x_{t-1}, u_t)$ must be a linear function

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

Process noise

$$\varepsilon_t \sim p_\varepsilon(\cdot) = \mathbb{N}(0, R_t)$$

x_t and x_{t-1} : state vectors

u_t : control vector

A_t : a matrix of size n x n

B_t : a matrix of size n x m

ε_t : Gaussian random vector that models the uncertainty (zero mean and covariance R_t)

$$x_t = \begin{bmatrix} x_{1,t} \\ x_{2,t} \\ \vdots \\ x_{n,t} \end{bmatrix} \qquad u_t = \begin{bmatrix} u_{1,t} \\ u_{2,t} \\ \vdots \\ u_{m,t} \end{bmatrix}$$

Kalman Filter

– Measurement probability $p(z_t \mid x_t)$ must also be linear

$$z_t = C_t x_t + \boxed{\delta_t}$$

Measurement noise

$$\delta_t \sim p_\delta(\cdot) = \mathbb{N}(0, Q_t)$$

z_t : measurement vector

C_t : a matrix of size k x n

δ_t : multi-variate Gaussian with zero mean and covariance Q_t

– The initial belief $bel(x_0)$ must be normally distributed with the mean μ_0 and covariance Σ_0

These three assumptions are sufficient to ensure that $bel(x_t)$
the posterior is always Gaussian for any point of time t

$\mathbb{N}(m, R)$: Gaussian random variable with mean m and variance R

Kalman Filter

- Algorithm

Algorithm Kalman_filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$)

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$$

Predicted belief $\overline{bel}(x_t)$

Kalman Gain

$$\boxed{K_t} = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t \boxed{(z_t - C_t \bar{\mu}_t)}$$

$$\Sigma_t = (I - K_t C_t)\bar{\Sigma}_t$$

Posterior belief $bel(x_t)$

Return μ_t, Σ_t

innovation

Innovation : output prediction error, measurement residual

Kalman Filter

- A brief summary

Linear-Gaussian system:

$$\begin{cases} x_t = A_t x_{t-1} + B_t u_t + \boxed{\varepsilon_t} \\ z_t = C_t x_t + \boxed{\delta_t} \end{cases}$$

Process noise

$$\varepsilon_t \sim p_\varepsilon(\cdot) = \mathbb{N}(0, R_t)$$

Measurement noise

$$\delta_t \sim p_\delta(\cdot) = \mathbb{N}(0, Q_t)$$

Assume:

$$bel(x_0) = \mathbb{N}(\mu_0, \Sigma_0)$$

Kalman Filter

- Prediction

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

Process noise

$$\varepsilon_t \sim p_\varepsilon(\cdot) = \mathbb{N}(0, R_t)$$

Motion model

$$p(x_t \mid x_{t-1}, u_t) = p_\varepsilon(x_t - A_t x_{t-1} - B_t u_t) = \mathbb{N}(A_t x_{t-1} + B_t u_t, R_t)$$

Predicted belief

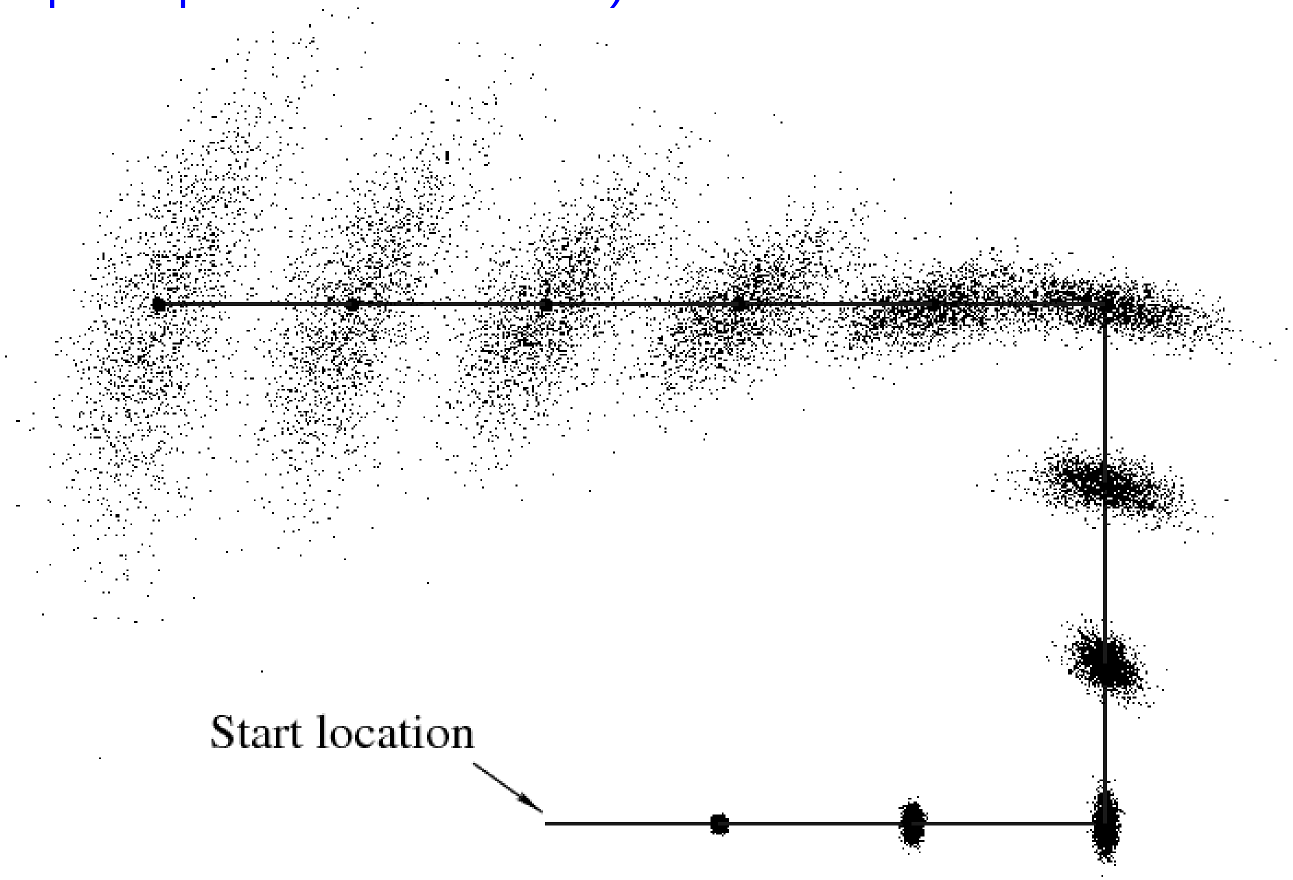
$$\overline{bel}(x_t) = \int_R \underbrace{p(x_t \mid x_{t-1}, u_t)}_{\mathbb{N}(A_t x_{t-1} + B_t u_t, R_t)} \; \underbrace{bel(x_{t-1})}_{\mathbb{N}(\mu_{t-1}, \Sigma_{t-1})} \, dx_{t-1}$$

The convolution of two Gaussian distributions → still Gaussian!

Prediction Step KF	$\overline{bel}(x_t) = \mathbb{N}(\bar{\mu}_t, \bar{\Sigma}_t)$	$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$ $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$

Kalman Filter

Accumulation of the pose estimation error based on the motion model (only proprioceptive measurements)



From Thrun Burgard Fox, Probabilistic Robotics, MIT Press 2006

Kalman Filter

- Measurement Update

$$z_t = C_t x_t + \boxed{\delta_t}$$

Measurement noise

$$\delta_t \sim p_\delta(\cdot) = \mathbb{N}(0, Q_t)$$

Measurement model

$$p(z_t \mid x_t) = p_\delta(z_t - C_t x_t) = \mathbb{N}(C_t x_t, Q_t)$$

Updated belief

$$bel(x_t) = \eta \underbrace{p(z_t \mid x_t)}_{\mathbb{N}(C_t x_t, Q_t)} \quad \underbrace{\overline{bel}(x_t)}_{\mathbb{N}(\bar{\mu}_t, \bar{\Sigma}_t)}$$

Correction Step KF

$$bel(x_t) = \mathbb{N}(\mu_t, \Sigma_t)$$

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

$$\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$$

$$\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$$

Kalman Filter

- Measurement Update

$$z_t = C_t x_t + \boxed{\delta_t}$$

Measurement noise

$$\delta_t \sim p_\delta(\cdot) = \mathbb{N}(0, Q_t)$$

Estimate x_t from the measurement z_t with minimum mean square error criteria

Estimated value is denoted by $\tilde{x}_t = K_t z_t$

Error is defined as $e_t = x_t - \tilde{x}_t = x_t - K_t z_t$

$$E[x_t x_t^{\,T}] = \bar{\Sigma}_t$$

$$E[\delta_t \delta_t^{\,T}] = Q_t$$

MMSE criteria $\hat{K}_t = \arg\min_{K_t} E[e_t e_t^{\,T}]$

$$E[e_t e_t^{\,T}] = E[(x_t - K_t C_t x_t - K_t \delta_t)(x_t - K_t C_t x_t - K_t \delta_t)^T]$$

$$\frac{dE[e_t e_t^{\,T}]}{dK_t^{\,T}} = 0 \qquad \frac{dE[e_t e_t^{\,T}]}{dK_t^{\,T}} = K_t C_t E[x_t x_t^{\,T}] C_t^{\,T} + K_t E[\delta_t \delta_t^{\,T}] - E[x_t x_t^{\,T}] C_t^{\,T}$$

$$\hat{K}_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$$

Kalman Filter

- A closer look

Previous estimate

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$$

E_{est}

$$K_t = \frac{\bar{\Sigma}_t C_t^T}{C_t \bar{\Sigma}_t C_t^T + Q_t}$$

E_{mea}

$$\mu_t = \bar{\mu}_t + K_t \boxed{(z_t - C_t \bar{\mu}_t)}$$

$$\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$$

innovation

EST_t : current estimate

EST_{t-1} : previous estimate

MEA : measurement

K_t : Kalman gain

E_{est} : Error in estimate

E_{mea} : Error in measurement

$$K_t = \frac{E_{est}}{E_{est} + E_{mea}} \qquad 0 \le K_t \le 1$$

$$EST_t = EST_{t-1} + K_t (MEA - EST_{t-1})$$

$$E_{est_t} = \frac{E_{mea}}{E_{est_{t-1}} + E_{mea}} E_{est_{t-1}} = (I - K_t) E_{est_{t-1}}$$

Kalman Filter

- A closer look

$$K_t = \frac{E_{est}}{E_{est} + E_{mea}} \qquad 0 \le K_t \le 1$$

$$EST_t = EST_{t-1} + K_t(MEA - EST_{t-1})$$

$$E_{est_t} = (I - K_t)E_{est_{t-1}}$$

K_t	Measurements	Estimates
1	accurate	unstable
0	inaccurate	stable

EST_t : current estimate

EST_{t-1} : previous estimate

MEA : measurement

K_t : Kalman gain

E_{est} : Error in estimate

E_{mea} : Error in measurement

Kalman Filter

- A closer look

Initial state

$$\mu_0 \qquad \Sigma_0$$

Previous state

$$\mu_{t-1} \qquad \Sigma_{t-1}$$

New state (predicted)

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$$

Compute Kalman gain

$$K_t = \frac{\bar{\Sigma}_t C_t^T}{C_t \bar{\Sigma}_t C_t^T + Q_t}$$

Update with measurement

$$\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$$

$$\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$$

Measurement

$$z_t = C_t x_t + \delta_t$$

Extended Kalman Filter (EKF)

- Extension to non-linear systems

Non-linear - Gaussian system:

$$\begin{cases} x_t = g(x_{t-1}, u_t + \boxed{\varepsilon_t}) \\ z_t = h(x_t) + \boxed{\delta_t} \end{cases}$$

Process noise

$$\varepsilon_t \sim p_\varepsilon(\cdot) = \mathbb{N}(0, R_t)$$

Measurement noise

$$\delta_t \sim p_\delta(\cdot) = \mathbb{N}(0, Q_t)$$

Assume:

$$bel(x_0) = \mathbb{N}(\mu_0, \Sigma_0)$$

The key idea of the EKF approximation is linearization.

EKF utilizes Taylor expansion for linearization.

Extended Kalman Filter (EKF)

- Linearization
  - Prediction

$$g(x_{t-1}, u_t) \approx g(\mu_{t-1}, u_t) + \underbrace{\frac{\partial g(\mu_{t-1}, u_t)}{\partial x_{t-1}}}_{=:G_t} (x_{t-1} - \mu_{t-1}) = g(\mu_{t-1}, u_t) + G_t(x_{t-1} - \mu_{t-1})$$

$\xrightarrow{\hspace{2em}}$ Jacobian

  - Correction

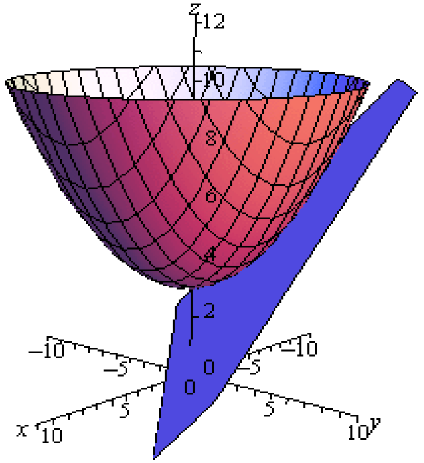
$$h(x_t) \approx h(\bar{\mu}_t) + \underbrace{\frac{\partial h(\bar{\mu}_t)}{\partial x_t}}_{=:H_t} (x_t - \bar{\mu}_t) = h(\bar{\mu}_t) + H_t(x_t - \bar{\mu}_t)$$

$\xrightarrow{\hspace{2em}}$ Jacobian

Extended Kalman Filter (EKF)

- Jacobian matrix
  - Typically $m \times n$ non-square matrix
  - Given a vector-valued function

$$g(x) = \begin{bmatrix} g_1(x) \\ \vdots \\ g_m(x) \end{bmatrix}$$



  - Jacobian matrix is defined as

$$G_t = \begin{bmatrix} \dfrac{\partial g_1}{\partial x_1} & \cdots & \dfrac{\partial g_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial g_m}{\partial x_1} & \cdots & \dfrac{\partial g_m}{\partial x_n} \end{bmatrix}$$

  - The orientation of the tangent plane to the vector-valued function at a given point
  - A generalization of the gradient of a scalar valued function

Extended Kalman Filter (EKF)

$$\begin{cases} x_t = g(x_{t-1}, u_t + \varepsilon_t) \\ z_t = h(x_t) + \delta_t \end{cases}$$

KF

Prediction Step

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$
$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$$

Correction Step

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

EKF

Prediction Step

$$\bar{\mu}_t = g(\mu_{t-1}, u_t)$$
$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + G_t R_t G_t^T$$

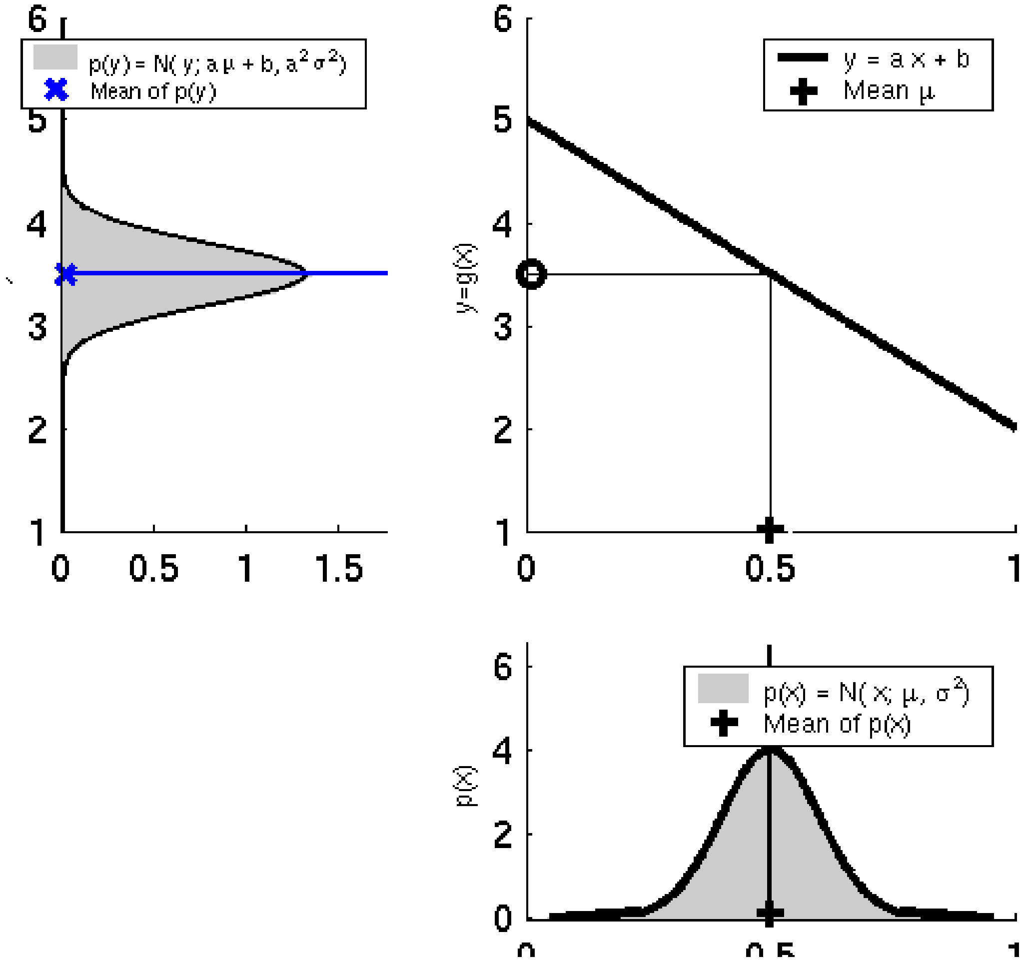
Correction Step

$$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$$
$$\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$$
$$\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$$

Jacobian

Kalman Filter (KF)

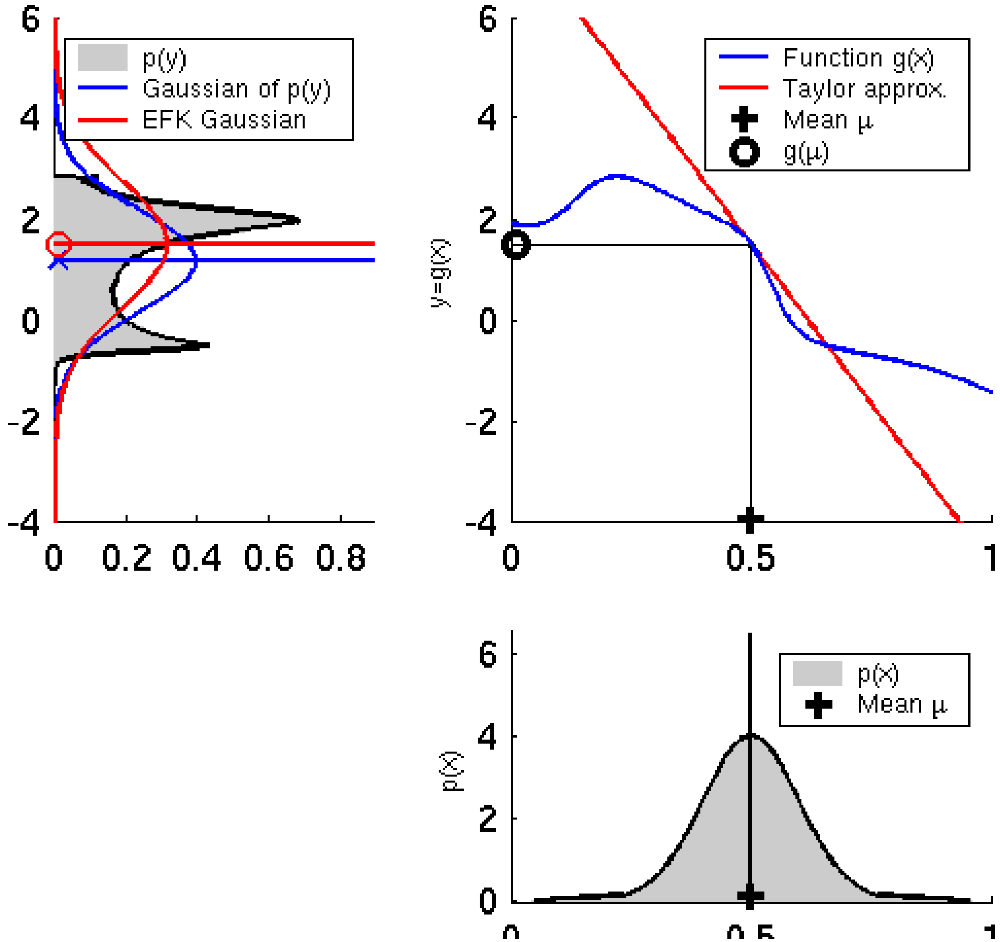
- Linear transformation of a Gaussian variable



From Thrun Burgard Fox, Probabilistic Robotics, MIT Press 2006

Extended Kalman Filter (EKF)

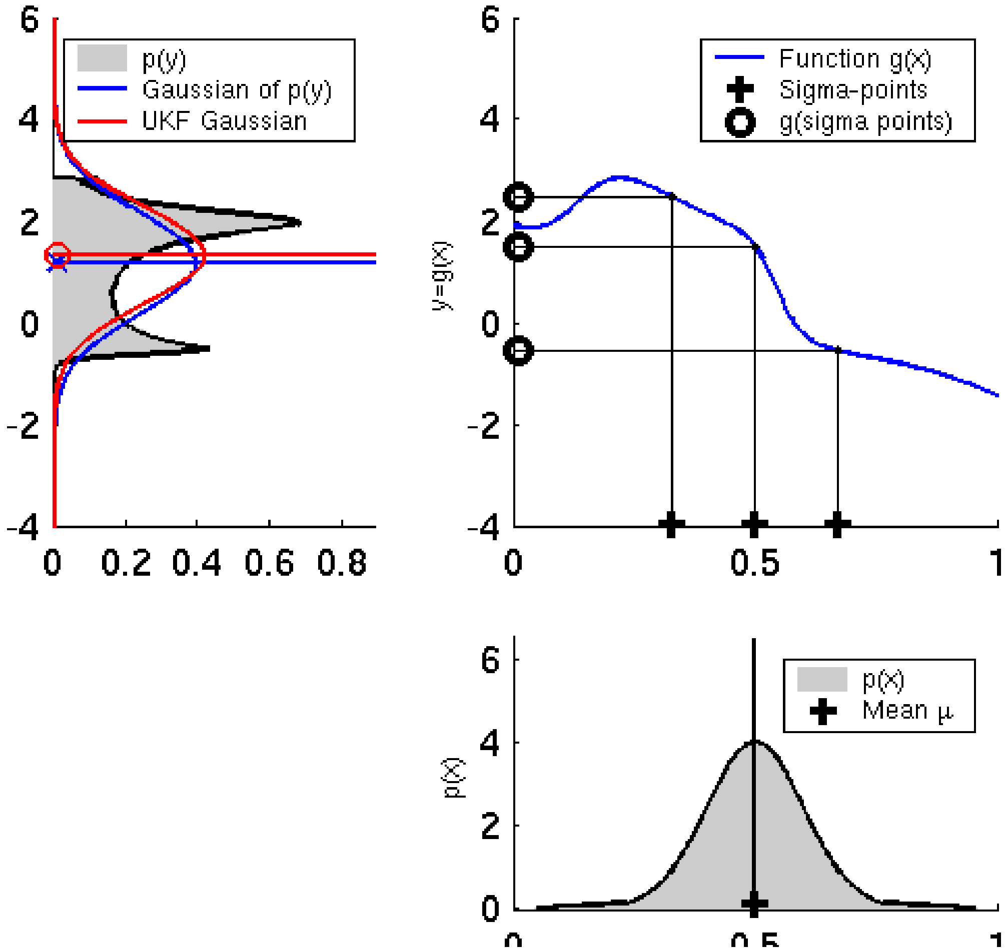
- Non-linear transformation of a Gaussian variable



From Thrun Burgard Fox, Probabilistic Robotics, MIT Press 2006

Unscented Kalman Filter (UKF)

- Non-linear transformation of a set of sigma points



From Thrun Burgard Fox, Probabilistic Robotics, MIT Press 2006

Kalman Filter

- Example 1: moving object in 1 dimension

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

$$z_t = C_t x_t + \delta_t$$

$$X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \qquad A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} \qquad U = [\, a \,]$$

$$AX = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} = \begin{bmatrix} x + \Delta t\, \dot{x} \\ \dot{x} \end{bmatrix} \qquad BU = \begin{bmatrix} \frac{1}{2}\Delta t^2\, a \\ \Delta t\, a \end{bmatrix}$$

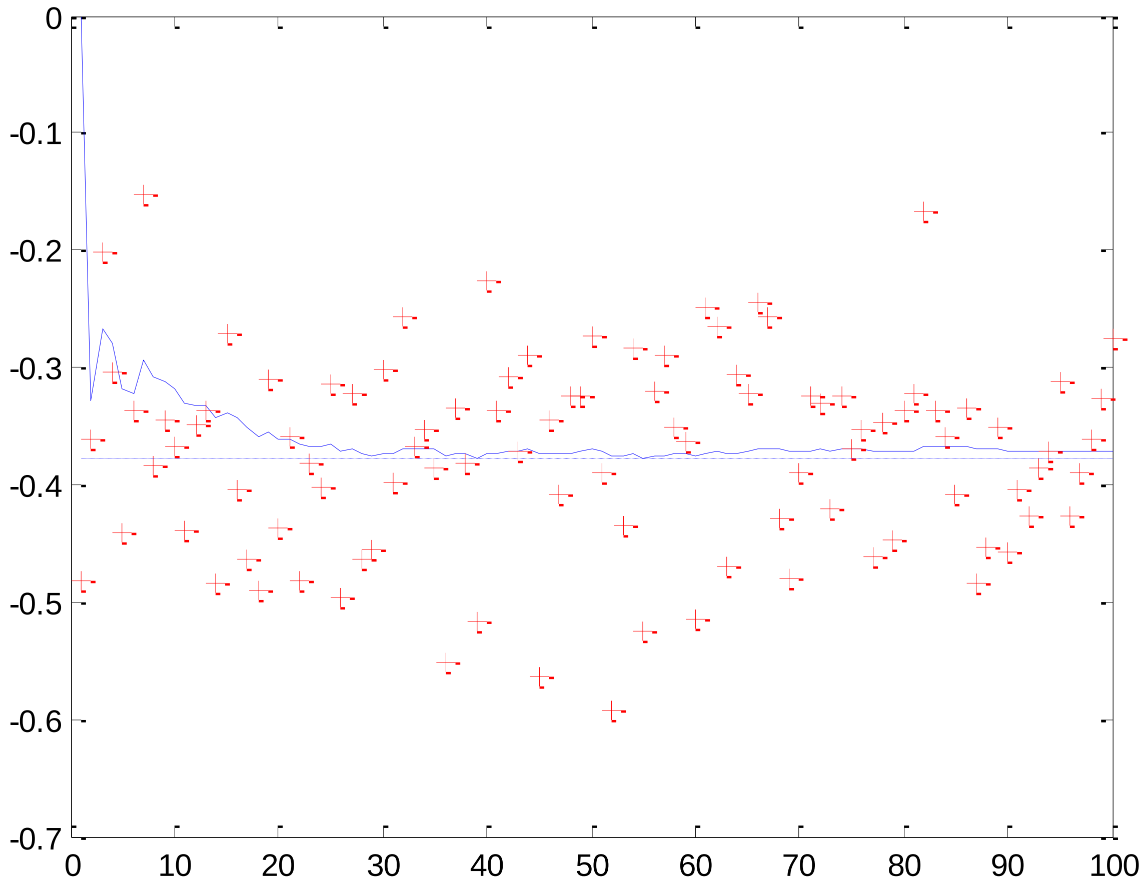
$$x_{t-1} = 20$$
$$\dot{x}_{t-1} = 2$$
$$\Delta t = 0.1$$
$$a = 1$$

$$X_t = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ \dot{x}_{t-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2\, a \\ \Delta t\, a \end{bmatrix}$$

$$= \begin{bmatrix} 20.2 \\ 2 \end{bmatrix} + \begin{bmatrix} 0.005 \\ 0.1 \end{bmatrix} = \begin{bmatrix} 20.205 \\ 2.1 \end{bmatrix}$$

Kalman Filter

- Example 1: moving object in 1 dimension



Kalman Filter

- Example 2: falling object

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

$$z_t = C_t x_t + \delta_t$$

$$X = \begin{bmatrix} y \\ \dot{y} \end{bmatrix} \qquad A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} \qquad U = [\, g \,]$$

$$AX = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} = \begin{bmatrix} y + \Delta t\, \dot{y} \\ \dot{y} \end{bmatrix} \qquad BU = \begin{bmatrix} \frac{1}{2}\Delta t^2\, g \\ \Delta t\, g \end{bmatrix}$$

$$y_{t-1} = 20$$

$$\dot{y}_{t-1} = 0$$

$$\Delta t = 0.1$$

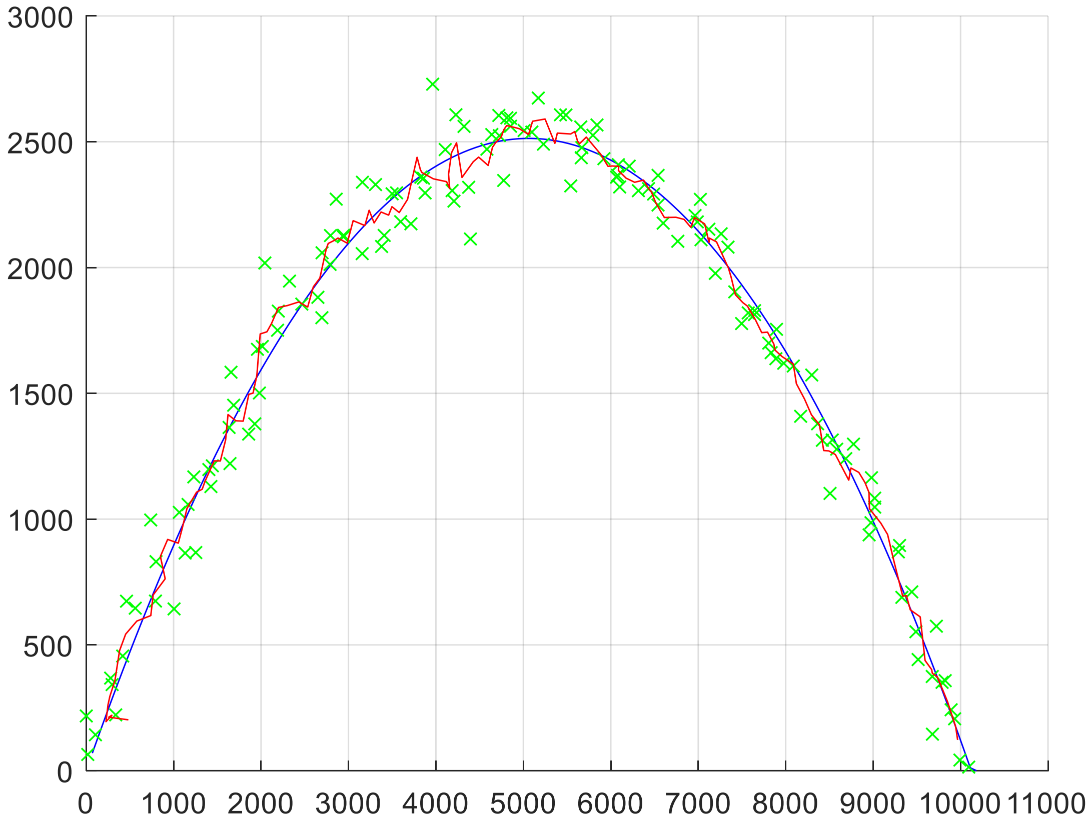
$$g = -9.8$$

$$X_t = \begin{bmatrix} y_{t-1} + \Delta t\, \dot{y}_{t-1} + \frac{1}{2}\Delta t^2\, g \\ \dot{y}_{t-1} + \Delta t\, g \end{bmatrix}$$

$$= \begin{bmatrix} 20 + 0.5 * (0.1)^2 * (-9.8) \\ -9.8 \end{bmatrix} = \begin{bmatrix} 20.151 \\ -0.98 \end{bmatrix}$$

Kalman Filter

- Example 2: falling object



Kalman Filter

- Example 3: moving object in 2 dimensions

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

$$z_t = C_t x_t + \delta_t$$

$$X = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} \qquad A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} U = \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix}$$

$$X_t = \begin{bmatrix} x_{t-1} + \Delta t\, \dot{x}_{t-1} + \frac{1}{2}\Delta t^2\, \ddot{x} \\ y_{t-1} + \Delta t\, \dot{y}_{t-1} + \frac{1}{2}\Delta t^2\, \ddot{y} \\ \dot{y}_{t-1} + \Delta t\, \ddot{x} \\ \dot{y}_{t-1} + \Delta t\, \ddot{y} \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Kalman Filter

- Example 3: moving object in 2 dimensions