So today we are going to talk abt state estimation

STATE ESTIMATION.

Today's outline is as follows: So first of all, we are going to ~~tat~~ start with a simple linear model, then we will see how far we can go to generalised it. Then in the second part, we are going to review the traditional way to derive KF. In the end we will focus on the Bayesian way of deriving KF. So we can somehow combine Bayes filter with KF.

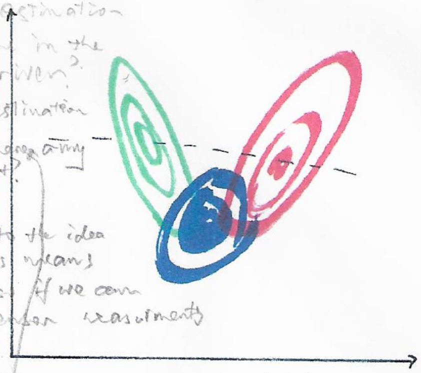
Motivation : Man in the woods problem:

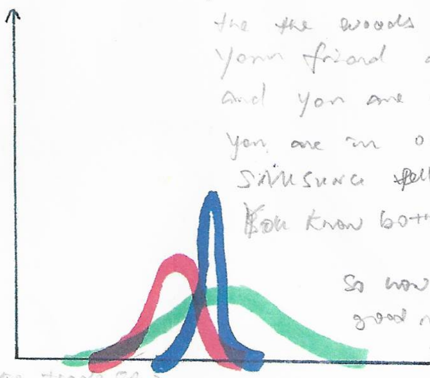
Device a and b provide independent estimate of hidden variable. (position on the map). Each device has Gaussian Noise property. The ellipse describes the region centered on the mean in the distribution that contains 10%, 25% and 50% of data under the distribution. The maximum likelihood estimate of the hidden variable is marked at the center.

Now, a new problem is what if your estimation tells you you are in the middle of the river? Obviously the estimation is off. So is there a any other way to do it?

⇓

This leads us to the idea that more inputs means better result. so if we can cooperate the sensor measurements in the past and recursively estimate the state we can somehow have better results





Let's start with a simple scenario, A man the the woods problem. Say you and your friend are walking in Wyman Park and you are lost. Your Iphone tells you you are in one location but your friend's Samsung tells you are in another location. You know both of the GPS... are off.

So now our motivation is " is there a good way that we can take a trade off of these two sensors"

Or is there a better way than just take trade off?

One typical approach is to find the ML of these two sensors by taking into account the quality of / noise of the sensor ... cheap → more noise → less likely

This is also Biological relevant because in your brain you have the capacity to rely on multiple sensors you have things touch, see...

Linear & Non Linear Process

why (PIPDS)

- Linear: ? In general, A and B can be time varying matrix and it can still be linear

$$x_{t+1} = Ax_t + Bu_t + w_t$$

$$z_t = Hx_t + v_t$$

$$w_t \sim N(0, Q_t)$$

$\begin{bmatrix} x^2 & \\ & y \end{bmatrix} \begin{pmatrix} w \\ y \end{pmatrix}$ not x, ~~or~~ only x is state space

- Non-Linear:

$$x_{t+1} = f(x_t, u_t, w_t)$$

$$z_t = h(x_t, v_t)$$

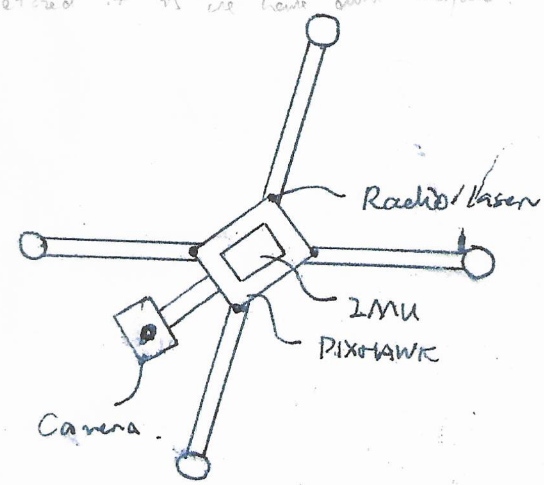
$$v_t \sim N(0, R_t)$$

These systems are derived in the following manner. So first we have a linear system, $\dot{x} = Ax + Bu$, which is a first order differential equation. The system is a ~~discrete~~ continuous time system so we can find the soln for that, Finally we can discretized it to be get a time update of the state vector.

For the system that is not nonlinear, we can ~~discreti~~ linearize it using Taylor series. Then discretized it as we have done before.

Let's follow the idea and start with a slightly more generalized model to see what we can have

Complex System Model: Mobile Robot



System model:

Since here you echo rostf, it will give you a quaternion ~~version~~, so here let's use quaternion

$$x_t = [q_w \; q_x \; q_y \; q_z \; \Delta x \; \Delta y \; \Delta z \; V_x \; V_y \; V_z]$$

The derivation for quaternion:

$$\dot{q}(t) = \frac{w(t)}{2} q(t) =$$

Remember that the ~~derivation~~ definition of linear system is the change of output is proportional to change of input. so we have $\dot{x} = Ax + Bu$ now let's find $\dot{x}$. For quaternion, we have a theorem that $\dot{q}(t) = \frac{w}{2} q(t)$, where w is the angular velocity determined by $q(t)$.

skew symmetric $[0,\ w_x\ w_y\ w_z]^T$

$$= \frac{1}{2}\begin{bmatrix} 0 & -w_x & -w_y & -w_z \\ w_x & 0 & w_z & -w_y \\ w_y & -w_z & 0 & w_x \\ w_z & w_y & -w_x & 0 \end{bmatrix}\begin{bmatrix} q_w \\ q_x \\ q_y \\ q_z \end{bmatrix} = \hat{\omega}\cdot q$$

The angular velocity can be found from Newton's equations for dynamics.

One thing to take note is ~~that~~ the only thing you measure here is the acceleration and angular rate. So for an IMU, it contains a accelerometer, which measures the acceleration and a gyroscope, which measure ~~the~~ angular rate, so these two measurements are in the body frame, But for velocity and position. They are in world frame

The robot Kinematics:

$$[\dot{\Delta x},\ \dot{\Delta y},\ \dot{\Delta z}]^T = [v_x,\ v_y,\ v_z]^T$$

The derivative of robot velocity is:

$$\begin{bmatrix} \dot{v}_x \\ \dot{v}_y \\ \dot{v}_z \end{bmatrix} = R_{wb}^{bw}\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \underbrace{g}_{\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}},\quad \text{where}\quad R_{wb}^{bw} = \begin{bmatrix} 1-2q_y^2-2q_z^2 & 2q_xq_y-2q_zq_w & 2q_xq_z+2q_yq_w \\ 2q_xq_y+2q_zq_w & 1-2q_x^2-2q_z^2 & 2q_yq_z-2q_xq_w \\ 2q_xq_z-2q_yq_w & 2q_yq_z+2q_xq_w & 1-2q_x^2-2q_y^2 \end{bmatrix}$$

for most of ~~sensors~~ your IMU doesn't give you the the specific force, so when you are at rest you still have this g in readings so you need to compensate for this g.

is the rotation matrix of quaternion from world to body.

Then the continuous system model:

$$\dot{x}_v = [\dot{q}, \dot{p}, \dot{v}] = Ax_v + Bu$$

$$\begin{bmatrix} \dot{p} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} p \\ v \end{bmatrix} = \begin{bmatrix} v \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ Ra+g \end{bmatrix} = \underbrace{\begin{bmatrix} \hat{\omega} & 0 & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix}}_{4\times 6}\underbrace{\begin{bmatrix} q \\ p \\ v \end{bmatrix}}_{10\times 1} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ R_w^b I g \end{bmatrix}}_{\substack{3\times 3 \quad 3\times 1 \\ 3\times 4}} \underbrace{\begin{bmatrix} a \\ 1 \end{bmatrix}}_{3\times 1}$$

so far we have the continuous-time linear system model, and we want to discretize it for implementation purpose ~~as~~ we are using digital electronics so the time is discrete.

~~If~~ For implementation, the system equation needs to be discretized by matrix exponential $Ad = e^{AT}$. For the first diagonal block. $(e^{\hat{\omega}t} = I + \hat{w}\sin t + \hat{w}^2(1-\cos\theta))$

Use Rodrigues's Formula we have $e^{\frac{W\Delta t}{2}} = \left[\begin{matrix} \cos\|W\|\Delta t/2 \\ \frac{W}{\|W\|}\sin(\|W\|\Delta t/2) \end{matrix}\right]$

recall that the solution for 1st order differential equation is (let the initial $t_0 = t_{k-1}$)

$$q_{k+1} = e^{\frac{1}{2}\int_0^T W dt} \cdot q_k = A_q \cdot q_k = \begin{bmatrix} r_0 & -r_1 & -r_2 & -r_3 \\ r_1 & r_0 & r_3 & -r_2 \\ r_2 & -r_3 & r_0 & r_1 \\ r_3 & r_2 & -r_1 & r_0 \end{bmatrix} \cdot q_k$$

put into skew symmetric form

$$x(t_k) = e^{A(t_k - t_{k-1})} x(t_{k-1}) + \int_{t_{k-1}}^{t_k} e^{A(t_k-\tau)} d\tau\, Bu(t_{k-1})$$

let ~~the~~ $\Delta t = t_k - t_{k-1}$, $A(\tau)\ B(\tau)\ u(\tau)$ to be constant during the interval, Where $r_0 = \cos\frac{\Delta t \|W\|}{2}$, $r_1 = \frac{\sin(\Delta t\|W\|/2)\,W_x}{\|W\|}$, $r_2 = \frac{\sin(\Delta t\|W\|/2)\,W_y}{\|W\|}$, $r_3 = \frac{\sin(\Delta t\|W\|/2)\,W_z}{\|W\|}$

$$\text{so } x(t_k) = e^{A\Delta t}x(t_{k-1}) + e^{A\Delta t}\int_0^{\Delta t} e^{-A\lambda} d\lambda\, Bu(t_{k-1})$$

The second diagonal block

$= F_{k-1}X_{k-1} + G_{k-1}U_{k-1}$

State transition matrix

$$\begin{bmatrix} \dot{\Delta p} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \Delta p \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ R_w^b I g \end{bmatrix}\begin{bmatrix} a \\ 1 \end{bmatrix}$$

SKIP

Then

$$\begin{bmatrix} \Delta p_{k+1} \\ v_{k+1} \end{bmatrix} = A_{pv}\begin{bmatrix} \Delta p_k \\ v_k \end{bmatrix} + B_{pv}u_k$$

where

$$A_{pv} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{6\times 6}, \qquad B_{pv} = \begin{bmatrix} T & 0 & 0 & \frac{\Delta T^2}{2} & 0 & 0 \\ 0 & T & 0 & 0 & \frac{\Delta T^2}{2} & 0 \\ 0 & 0 & T & 0 & 0 & \frac{\Delta T^2}{2} \\ 0 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 0 & 0 & T \end{bmatrix} \cdot \begin{bmatrix} 0_{3\times 4} \\ R^b_w \, | \, g \end{bmatrix}^{6\times 4}$$

3×4

$$u_k = [a_x \;\; a_y \;\; a_z \;\; 1]^T$$

In the end, the discretized system equation can be written as

$$x_{k+1} = A_d(k) \cdot \begin{bmatrix} q_k \\ [AP_k^T, \, v_k^T]^T \end{bmatrix} + B_k u_k$$

$$= \begin{bmatrix} A_q & 0 \\ 0 & pv \end{bmatrix} \begin{bmatrix} q_k \\ [AP_k^T, \, v_k^T]^T \end{bmatrix} + \begin{bmatrix} 0 \\ B_{pv} \end{bmatrix} u_k$$

$$\begin{bmatrix} x \\ y \\ z \\ V_x \\ V_y \\ V_z \end{bmatrix} = \begin{bmatrix} x + V_x \Delta t \\ y + V_y \Delta t \\ z + V_z \Delta t \\ V_x \\ V_y \\ V_z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ (R_{11}+R_{12}+R_{13}) \frac{a_x}{2} \Delta t^2 \\ (R_{21}+R_{22}+R_{23}) \frac{a_y}{2} \Delta t^2 \\ (R_{31}+R_{32}+R_{33}) \frac{a_z}{2} \Delta t^2 \end{bmatrix}$$

Measurement model:

- Laser / radio : L2-Norm model with Gaussian white noise

$$z_l^i = \| [x \; y \; z]^T - [x_i \; y_i \; z_i]^T \| + n_l \sim N(0, \sigma_l^2)$$

or 2D - range bearing

$$\begin{bmatrix} \| [x \;\; y]^T - [x_i \;\; y_i]^T \| \\ \tan^{-1}\left(\frac{y - y_i}{x - x_i}\right) - \theta_t \end{bmatrix}$$

- Camera model :

$$z_c^i = K_c \cdot T_w^c \cdot [x_i \; y_i \; z_i \; 1]^T + n_c \sim N(0, \sigma_c^2)$$

where

$$K_c = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \qquad T_w^c = [R \;\; t]$$

focal length in terms of pixel

skew coefficient

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & \gamma & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \underbrace{[R \quad T]}_{3\times 4} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}^{4\times 1}$$

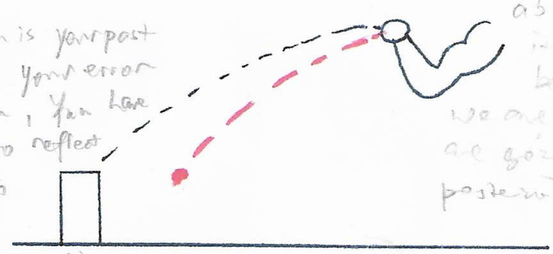
$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$$

Kalman Filtering : A Frequentist Approach

Let's come back to Kalman filter. so just now we have talked about the variance or uncertainty of each sensor which comes from the sensor noise. So the Kalman filter tells us to estimate a parameter in such a way that we minimize our uncertainty about it after we altered our belief about it. So we are going to have some prior belief abt what this parameter is (location). We are going to make some measurements and we are going to combine this two things to form a posterior belief in a way to minimize our uncertainty.

Basic Learning Equation:

What Kalman says is that x'_k which is your past is a piece of information, and your error is a second piece of information, you have to combine them by set K to reflect the uncertainty you have. So by knowing this uncertainty you can weigh the two infor.



sensitivity to error

Kalman was saying that when you are trying to do your estimation, you have all the past and you have this current data point. If you talk abt Gaussian Noise, all the past are in two parameters, mean and variance. The current estimate can be incorporated taken into account the uncertain of the parameter

$$x_k = \underbrace{x'_k}_{\text{prior}} + K_k(\underbrace{z_k}_{\text{observe}} - \underbrace{H_k x'_k}_{\text{predict}})$$

posterior

$$= (I - K_k H) x'_k + K_k z_k$$

$$z = Hx + R\varepsilon \sim N(0,R)$$

to the past by

Kalman Says in addition to have this estimates which really are the mean of your distribution. You also have a variance associated with it.

Similarly we have the covariance to be :

$$\Sigma_k = (I - K_k H)\Sigma'_k (I - K_k H)^T + K_k R K_k^T$$

R is the only variable in z

R to be the measurement covariance

$$= \Sigma'_k - K_k H_k \Sigma'_k - \Sigma'_k H_k^T K_k^T + K_k H_k \Sigma'_k H_k^T K_k^T + K_k R_k K_k^T$$

So FIND the PARAM K that MINIMIZE THE VARIANCE

To find the minimum of covariance, use the trace. So in principle Σ_k is a covariance structure in which the diagonal element are the variance of individual element

$$tr[\Sigma_k] = tr[\Sigma'_k] - 2tr[K_k H_k^T \Sigma'_k] + tr[\underbrace{(H_k^T \Sigma'_k H_k + R_k)}_{\text{scalar}} K_k^T K_k^T]$$

$1\times n \; n\times n \; n\times 1$

transpose for Transpose

$$P = \begin{bmatrix} \sigma_1^2 & & \\ & \sigma_2^2 & \\ & & \ddots \; \sigma_n^2 \end{bmatrix}$$

$$\frac{\partial\, tr[AXB]}{\partial x} = A^T B^T$$

$$= tr[\Sigma'_k] - 2tr[K_k H \Sigma'_k] + (H_k \Sigma'_k H_k^T + R_k)\, tr[K_k^T K_k]$$

Norm: 1) positive

2) = 0 when that quantity is zero

$$= \cancel{tr[\Sigma'_k]} - 2H_k \Sigma'^T_k K_k + (H_k^T \Sigma'_k H_k + R) K_k^T K_k$$

Trace satisfies the property

$$tr[KK^T] = tr\begin{bmatrix} k_1^2 & & \\ & k_2^2 & \\ & & k_n^2 \end{bmatrix}$$

$$K^T K = [k_1, \dots, k_n] \begin{bmatrix} k_1 \\ \vdots \\ k_n \end{bmatrix}$$

$1\times n \quad n\times 1$

Take derivative w.r.t K and set it to zero

$$\frac{d\, tr[\Sigma_k]}{dK_k} = -2\Sigma'_k H_k^T + 2(H_k \Sigma'_k H_k^T + R_k) K_k = 0$$

$$K = \Sigma'_k H_k^T \cdot (H_k \Sigma'_k H_k^T + R_k)^{-1}$$

So the Kalman gain we are computing is the input H_k and the history of our covariance, so I keep the history of what I have seen, and incorporate it into the current estimation

Finally

$$x_k = x'_k + K_k(z_k - H_k x'_k)$$

$$\Sigma_k = \Sigma'_k \underbrace{(I - K_k H_k)}$$

the posterior covariance is the prior covariance multiply something less than 1. So your covariance is getting smaller

Kalman Filtering: A Bayesian Approach

Recall Bayes Theorem:

$$P(x|z) = \frac{P(z|x)\,P(x)}{P(z)}$$

The posterior probability distribution can be written as:

$$P(x_k | u_{1:k}, z_{1:k}) = \frac{P(z_k | x_k, u_{1:k}, z_{1:k-1})\, P(x_{k}, u_{1:k}, z_{1:k-1})}{P(z_k | u_{1:k}, z_{1:k-1})}$$

Belief posterior

(current observation doesn't depend on $z_{1:k-1}, u_{1:k}$)

$$= \eta_k P(z_k | x_k, z_{1:k-1}, u_{1:k}) \cdot P(x_k | z_{1:k-1}, u_{1:k})$$

first term

$$= \eta_k P(z_k | x_k) \cdot P(x_k | z_{1:k-1}, u_{1:k})$$ // Markov Assumption. Given you know the state of the world, you can forget what happened at the past

missing last observation

recursive

Law of total prob

$$= \eta_k P(z_k | x_k) \int_{x_{k-1}} P(x_k | x_{k-1}, z_{1:k-1}, u_{1:k}) \cdot P(x_{k-1} | z_{1:k-1}, u_{1:k})\, dx_{k-1}$$

Now · Markov assumption

Current belief depends on control input and last bel, not current observation

$$= \eta_k P(z_k | x_k) \int_{x_{k-1}} P(x_k | u_k, x_{k-1})\, \boxed{P(x_{k-1} | u_{1:k-1}, z_{1:k-1})}\, dx_{k-1}$$

observation correction — innovation; transition; priori; belief at $t-1$; prediction

Current input doesn't help with current state. We don't care about the command that move the robot to the future

~~Simplify the notation:~~

This doesn't make sense. Don't need $P(x_k)$ this part

$$\cancel{= \eta P(z_k | x_k) \cdot \int P(z|x) \cdot P(x_{k-1})\, dx_{k-1}}$$

$$\cancel{= \eta P(z_k | x_k) \cdot P'(x_k)}$$

While let's keep it)

The prediction for ~~priori~~ posterior belief is also Gaussian.

Gaussian is closed under linear operations

$$P(A) = \int_B P(A|B) \cdot P(B)$$

$$P(x_k | u_k, x_{k-1}) = \mathcal{N}(x_k; A_k x_k + B_k u_k, Q_k)$$

$$P'(x_k) = \int P(x_k | u_k, x_{k-1}) \cdot P(x_{k-1})\, dx_{k-1}$$

How do I explain this can be gaussian

$$\cancel{= \int \eta \exp\{-\tfrac{1}{2}(x_k - A_k x_{k-1} - B_k u_k)^T Q_k^{-1} (x_k - A_k x_{k-1} - B_k u_k)\}}$$

$$\cancel{\cdot \exp\{-\tfrac{1}{2}(x_{k-1} - u_{k-1})^T \Sigma_{k-1}^{-1} (x_{k-1} - u_{k-1})\}\, dx_{k-1}}$$

So the prediction is a New Gaussian with (u'_t, Σ'_t)

$$P'(x_k) = \begin{cases} u'_k = A_k u_{k-1} + B_k u_k \\ \Sigma'_k = A_k \Sigma_{k-1} A^T + Q_k \end{cases}$$

The observation model is also Gaussian. So the posteriori can be written as:

$$P(x_k) = \eta \cdot P(z_k \mid x_k) \cdot P'(x_k)$$

$$= \eta \cdot \exp\{-\tfrac{1}{2}(z_k - H_k x_k)^T R_k^{-1} (z_k - H_k x_k)\} \cdot \exp\{-\tfrac{1}{2}(x_k - \mu'_k)^T \Sigma_k'^{-1} (x_k - \mu'_k)\}$$

To find the parameter x_k that maximize the posteriori

$$\arg\max_{x_k} = \arg\max_{x_k}(\eta \exp\{-\tfrac{1}{2}(z_k - H_k x_k)^T R_k^{-1}(z_k - H_k x_k)\} \cdot \exp\{-\tfrac{1}{2}(x_k - \mu'_k) \Sigma_k'^{-1} (x_k - \mu'_k)\}$$

take the negative sign out, and exponential out.

$$= \arg\min[(z_k - H_k x_k)^T R_k^{-1} (z_k - H_k x_k) + (x_k - A_k \mu_{k-1})^T (A_k \Sigma_{k-1} A_k^T + Q_k)^{-1} (x_k - A_k \mu_k)$$

Take the derivative w.r.t x_k and set it to zero

$$0 = 2(H_k^T R^{-1} H_k + (A_k \Sigma_{k-1} A_k^T + Q_k)^{-1}) x_k - 2(H_k R_k^{-1} z_k + (A \Sigma_{k-1} A^T + Q_k)^{-1} \cdot A \mu_{k-1})$$

$(\Sigma')^{-1} = \Sigma^{-1}$

$\Sigma_k = (I - KH)\Sigma'_k$

if $\Sigma'^{0} = \infty$, Σ''

$$x_k = [H_k^T R_k^{-1} H_k + \Sigma'_k]^{-1} [H_k^T R_k^{-1} H_k + \Sigma_k'^{-1} x'_k]$$

no prior belief →

"$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1} VA^{-1}$"

prior uncertainty ∞

$$= [\Sigma'_k - \Sigma'_k (R_k + H_k \Sigma'_k H_k^T)^{-1} H_k \Sigma'_k][H_k^T R_k^{-1} z_k + \Sigma_k'^{-1} x'_k]$$

Woodbury matrix identity

⇓

compute the Kalman Gain even if we have no prior belief.

$$= x'_k - K_k H_k x'_k + [\Sigma'_k H_k^T R^{-1} - \Sigma'_k H_k^{-1} (R_k + H_k \Sigma'_k H_k^T)^{-1} H_k \Sigma'_k H_k R^{-1}] z_k$$

$$= x'_k - K_k H_k x'_k + K_k[\underbrace{(H_k \Sigma'_k H_k^T + R_k) R_k^{-1}}_{I} \overset{\text{cancel}}{} - \overset{\text{out}}{\cancel{H_k \Sigma'_k H_k^T R_k^{-1}}}] z_k$$

⇓

ML, by taking the evidence,

$$= x_k - K_k H_k x'_k + K_k z_k$$

(Kalman Gain is going to be the ML estimate)

$$= x'_k - K_k(z_k - H_k x'_k)$$

If I start with knowing nothing,

Where

after I get my first sample,

I end up with $C^T R^{-1} C$

$$K_k = \Sigma'_k H_k^T (R_k + H_k \Sigma'_k H_k^T)^{-1}$$

And the posterior covariance

$$E[x_k \cdot x_k^T] = E[(x'_t + K_k z_k - K_k A_k x'_k)(x'_t + K_k z_k - K_k A_k x'_k)^T]$$

$$= (I - K_k H_k)\Sigma'_{k-1}(I - K_k H_k)^T + K_k R_k K_k^T$$

$$= \Sigma'_k - K_k H_k \Sigma'_k$$

Thanks Drs. Leonard & Hayes for the support!