Recursive Least Squares

<u>Model</u>

$$y_i = C_i x + e_i \quad , \; i = 1, 2, \dots$$

$y_i \in \mathbb{R}^m$, $x \in \mathbb{R}^n$, $e_i \in \mathbb{R}^m$, $C_i = m \times n$ real.

i - represents time index

x - an unknown **constant** vector

y_i = measurements

e_i = model "mismatch"

<u>Objective 1</u>: Compute a least squared error estimate of x at time k, using all available data,

$y_1, \dots, y_k$

<u>Objective 2</u> Discover a computationally attractive form for the answer.

<u>Solution</u>

$$\hat{x}_k := \arg\min_x \left(\sum_{i=1}^{k} (y_i - C_i x)^T S_i (y_i - C_i x) \right)$$

$$= \arg\min_x \left(\sum_{i=1}^{n} e_i^T S_i e_i \right)$$

where S_i = $m \times m$ positive def. matrix

$(S_i > 0)$

$$Y_k := \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix}, \quad A_k = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_k \end{bmatrix}, \quad E_k = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_k \end{bmatrix}$$

$$R_k = \mathrm{diag}(S_1, S_2, \dots, S_k) > 0$$

$$Y_k = A_k x + E_k$$

$$\|Y_k - A_k x\|^2 = \|E_k\|^2 := E_k^T R_k E_k$$

$$\hat{x}_k := \arg\min_x \|Y_k - A_k x\| \text{ satisfies}$$

the Normal Equations

$$\boxed{(A_k^T R_k A_k)\hat{x}_k = A_k^T R_k Y_k}$$

$$\therefore \quad \boxed{\hat{x}_k = (A_k^T R_k A_k)^{-1} A_k^T R_k Y_k}$$

This is called a Batch Solution.

<u>Drawback</u> $A_k = km \times n$ matrix, and grows at each step!

<u>Solution</u> Find a recursive means to compute $\hat{x}_{k+1}$ in terms of $\hat{x}_k$ and the new measurement y_{k+1}. Do

Normal equation at time k

$$A_k^T R_k A_k \hat{x}_k = A_k^T R_k Y_k$$

is equivalent to

$$\left(\sum_{i=1}^{k} C_i^T S_i C_i\right)\hat{x}_k = \sum_{i=1}^{k} C_i^T S_i y_i$$

We define

$$Q_k = \sum_{i=1}^{k} C_i^T S_i C_i$$

so that

$$Q_{k+1} = Q_k + C_{k+1}^T S_{k+1} C_{k+1}$$

At time k+1,

$$\left(\sum_{i=1}^{k+1} C_i^T S_i C_i\right)\hat{x}_{k+1} = \sum_{i=1}^{k+1} C_i^T S_i y_i$$

OR

$$Q_{k+1}\hat{x}_{k+1} = \underbrace{\sum_{i=1}^{k} C_i^T S_i y_i}_{Q_k \hat{x}_k} + C_{k+1}^T S_{k+1} y_{k+1}$$

$$\therefore \boxed{Q_{k+1}\hat{x}_{k+1} = Q_k\hat{x}_k + C_{k+1}^T S_{k+1} y_{k+1}}$$

Good start! Estimate at time k+1 expressed as a linear combination of the estimate at time k and the latest measurement at time k+1.

Continuing,

$$\hat{x}_{k+1} = Q_{k+1}^{-1}\left[Q_k \hat{x}_k + C_{k+1} S_{k+1} y_{k+1} \right]$$

Because $Q_k = Q_{k+1} - C_{k+1}^T S_{k+1} C_{k+1}$,

we have

$$\hat{x}_{k+1} = \hat{x}_k + \underbrace{Q_{k+1}^{-1} C_{k+1}^T S_{k+1}}_{\text{Kalman Gain}} \underbrace{(y_{k+1} - C_{k+1}\hat{x}_k)}_{\text{Innovations}}$$

Innovations $y_{k+1} - C_{k+1}\hat{x}_k$ = Measurement at time $k+1$ minus the "predicted" value of the measurement = "new information".

In a real-time implementation, computing the inverse of Q_{k+1} can be time consuming. An attractive alternative can be ~~achieved~~ obtained by applying the Matrix Inversion Lemma:

$$Q_{k+1}^{-1} = \left(Q_k + C_{k+1}^T S_{k+1} C_{k+1}\right)^{-1}$$

$$(A+BCD)^{-1} = A^{-1} - A^{-1}B\left(DA^{-1}B + C^{-1}\right)^{-1} DA^{-1}$$

$A \leftrightarrow Q_k \qquad B \leftrightarrow C_{k+1}^T \qquad C \leftrightarrow S_{k+1} \qquad D \leftrightarrow C_{k+1}$

$$Q_{k+1}^{-1} = Q_k^{-1} - Q_k^{-1} C_{k+1}^T \left[C_{k+1} Q_k^{-1} C_{k+1}^T + S_{k+1}^{-1}\right]^{-1} C_{k+1} Q_k^{-1}$$

Which is a recursion for Q_k^{-1}!

Upon defining

$$P_k = Q_k^{-1}$$

we have

$$P_{k+1} = P_k - P_k C_{k+1}^T \left[C_{k+1} P_k C_{k+1}^T + S_{k+1}^{-1} \right]^{-1} C_{k+1} P_k$$

We note that we are now inverting a matrix that is $m \times m$, instead of one that is $n \times n$. Typically, $n > m$, sometimes by a lot!