Rob 501 Handout: Grizzle
Recursive Least Squares

Model:

$$y_i = C_i x + e_i, \;\; i = 1, 2, 3, \cdots$$

$$\begin{aligned} C_i &\in \mathbb{R}^{m \times n} \\ i &= \text{time index} \\ x &= \text{an unknown } \underline{\text{constant}} \text{ vector} \in \mathbb{R}^n \\ y_i &= \text{measurements} \in \mathbb{R}^m \\ e_i &= \text{model "mismatch"} \in \mathbb{R}^m \end{aligned}$$

Objective 1: Compute a least squared error estimate of x at time k, using all available data at time k, $(y_1, \cdots, y_k)$!

Objective 2: Discover a computationally attractive form for the answer.

Solution:

$$\begin{aligned} \hat{x}_k :&= \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \left(\sum_{i=1}^{k} (y_i - C_i x)^\top S_i (y_i - C_i x) \right) \\ &= \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \left(\sum_{i=1}^{k} e_i^\top S_i e_i \right) \end{aligned}$$

where $S_i = m \times m$ positive definite matrix. ($S_i > 0$ for all time index i)

Batch Solution:

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

$$R_k = \begin{bmatrix} S_1 & & & \mathbf{0} \\ & S_2 & & \\ & & \ddots & \\ \mathbf{0} & & & S_k \end{bmatrix} = diag(S_1, S_2, \cdots, S_k) > 0$$

$Y_k = A_k x + E_k$, [model for $1 \leq i \leq k$]
$\|Y_k - A_k x\|^2 = \|E_k\|^2 := E_k^\top R_k E_k$

Since $\hat{x}_k$ is the value minimizing the error $\|E_k\|$, which is the unexplained part of the model,

$$\hat{x}_k = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \|E_k\| = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \|Y_k - A_k x\|,$$

which satisfies the Normal Equations $(A_k^\top R_k A_k)\hat{x}_k = A_k^\top R_k Y_k$.

$\therefore \underline{\hat{x}_k = (A_k^\top R_k A_k)^{-1} A_k^\top R_k Y_k}$, which is called a Batch Solution.

Drawback: $A_k = km \times n$ matrix, and grows at each step!

Solution: Find a recursive means to compute $\hat{x}_{k+1}$ in terms of $\hat{x}_k$ and the new measurement y_{k+1}!

Normal equations at time k, $(A_k^\top R_k A_k)\hat{x}_k = A_k^\top R_k Y_k$, is equivalent to

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

We define

$$M_k = \sum_{i=1}^{k} C_i^\top S_i C_i$$

so that

$$M_{k+1} = M_k + C_{k+1}^\top S_{k+1} C_{k+1}.$$

At time $k+1$,

$$\underbrace{(\sum_{i=1}^{k+1} C_i^\top S_i C_i)}_{M_{k+1}} \hat{x}_{k+1} = \sum_{i=1}^{k+1} C_i^\top S_i y_i$$

or

$$M_{k+1}\hat{x}_{k+1} = \underbrace{\sum_{i=1}^{k} C_i^\top S_i y_i}_{M_k\hat{x}_k} + C_{k+1}^\top S_{k+1} y_{k+1}.$$

$\therefore M_{k+1}\hat{x}_{k+1} = M_k\hat{x}_k + C_{k+1}^\top S_{k+1} y_{k+1}$

Good start on recursion! 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,

$$\hat{x}_{k+1} = M_{k+1}^{-1}\left[M_k\hat{x}_k + C_{k+1}^\top S_{k+1} y_{k+1}\right].$$

Because

$$M_k = M_{k+1} - C_{k+1}^\top S_{k+1} C_{k+1},$$

we have

$$\hat{x}_{k+1} = \hat{x}_k + \underbrace{M_{k+1}^{-1} C_{k+1}^\top S_{k+1}}_{\text{Kalman gain}} \underbrace{(y_{k+1} - C_{k+1}\hat{x}_k)}_{\text{Innovations}}.$$

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 M_{k+1} can be time consuming. An attractive alternative can be obtained by applying the Matrix Inversion Lemma:

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

Now, following the substitution rule as shown below,

$$A \leftrightarrow M_k \quad B \leftrightarrow C_{k+1}^\top \quad C \leftrightarrow S_{k+1} \quad D \leftrightarrow C_{k+1},$$

we can obtain that

$$\begin{aligned} M_{k+1}^{-1} &= \left(M_k + C_k^\top S_{k+1} C_{k+1}\right)^{-1} \\ &= M_k^{-1} - M_k^{-1} C_{k+1}^\top \left[C_{k+1} M_k^{-1} C_{k+1}^\top + S_{k+1}^{-1}\right]^{-1} C_{k+1} M_k^{-1}, \end{aligned}$$

which is a recursion for M_k^{-1}!

Upon defining

$$P_k = M_k^{-1},$$

we have

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

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!