ME 233 Advance Control II

Lecture 18

Least Squares Parameter Estimation

Least Squares Estimation

Model

$$y(k) = \sum_{i=1}^{n} \phi_i(k-1)\,\theta_i$$

Where

- $y(k)$ observed output
- $\phi_i(k)$ **known** and measurable function
- θ_i **unknown** but constant parameter

Least Squares Estimation

Model

$$y(k) = \phi^T(k-1)\,\theta$$

Where

$y(k)$ measured output

$$\underbrace{\phi(k) = \begin{bmatrix} \phi_1(k) \\ \vdots \\ \phi_n(k) \end{bmatrix}}_{n \times 1 \text{ regressor}} \qquad \underbrace{\theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}}_{\text{unknown vector}}$$

Batch Least Squares Estimation

Assume that we have collected k data sets:

$$\left.\begin{array}{l} y(1), \cdots, y(k) \\ \phi(0), \cdots, \phi(k-1) \end{array}\right\} \textit{collected data}$$

We want to find the parameter estimate at instant k: $\hat{\theta}(k)$

That best fits <u>all collected </u>data in the <u>**least squares** </u>sense:

$$\min_{\hat{\theta}(k)} \left\{ \frac{1}{2} \sum_{j=1}^{k} \left[y(j) - \phi^T(j-1)\,\hat{\theta}(k) \right]^2 \right\}$$

kept constant in the summation

Batch Least Squares Estimation

Defining the cost functional

$$V(\widehat{\theta}(k)) = \frac{1}{2}\sum_{j=1}^{k}\left[y(j) - \phi^T(j-1)\,\widehat{\theta}(k)\right]^2$$

$\widehat{\theta}(k)$ is obtained by solving

$$\frac{dV(\widehat{\theta}(k))}{d\widehat{\theta}(k)} = 0$$

Batch Least Squares Solution

The least squares parameter estimate $\widehat{\theta}(k)$ which solves

$$\frac{dV(\widehat{\theta}(k))}{d\widehat{\theta}(k)} = 0$$

Satisfies the **normal equation:**

$$\underbrace{\left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right]}_{n\times n \text{ matrix}}\widehat{\theta}(k) = \underbrace{\sum_{i=1}^{k}\phi(i-1)\,y(i)}_{n\times 1 \text{ vector}}$$

Normal Equation Derivation

$$V(\widehat{\theta}(k)) = \frac{1}{2}\sum_{j=1}^{k}\left[y(j) - \phi^T(j-1)\,\widehat{\theta}(k)\right]^2$$

$$\underbrace{\begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(k)\end{bmatrix}}_{Y(k)} - \underbrace{\begin{bmatrix} \phi_1(0) & \cdots & \phi_n(0) \\ \phi_1(1) & \cdots & \phi_n(1) \\ \cdot & \cdots & \cdot \\ \phi_1(k-1) & \cdots & \phi_n(k-1)\end{bmatrix}}_{\Phi^T(k-1)}\underbrace{\begin{bmatrix} \widehat{\theta}_1(k) \\ \vdots \\ \widehat{\theta}_n(k)\end{bmatrix}}_{\widehat{\theta}(k)}$$

$$V(\widehat{\theta}(k)) = \frac{1}{2}\left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]^T \left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]$$

Normal Equation Derivation

Parameter estimate after k observations: $\widehat{\theta}(k)$

$$V(\widehat{\theta}(k)) = \frac{1}{2}\left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]^T \left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]$$

Where

- $Y(k) = \left[\ y(1) \ \cdots \ y(k)\ \right]^T \in \mathcal{R}^k$
- $\Phi(k-1) = \left[\ \phi(0)\cdots\phi(k-1)\ \right] \in \mathcal{R}^{n\times k}$

Normal Equation Derivation

$$\Phi(k-1) = \left[\, \phi(0)\cdots\phi(k-1) \,\right] \in \mathcal{R}^{n\times k}$$

$$= \begin{bmatrix} \phi_1(0) & \cdots & \phi_1(k-1) \\ \phi_2(0) & \cdots & \phi_2(k-1) \\ \cdot & \cdots & \cdot \\ \cdot & \cdots & \cdot \\ \phi_n(0) & \cdots & \phi_n(k-1) \end{bmatrix}$$

Normal Equation Derivation

$\widehat{\theta}(k)$: Parameter estimate which solves

$$\frac{\partial V(\widehat{\theta}(k))}{\partial\widehat{\theta}(k)} = 0$$

$$V(\widehat{\theta}(k)) = \frac{1}{2}\left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]^T \left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]$$

$$\frac{\partial V(\widehat{\theta}(k))}{\partial\widehat{\theta}(k)} = -\left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right]^T \Phi^T(k-1)$$

$$= -\Phi(k-1)\left[Y(k) - \Phi^T(k-1)\,\widehat{\theta}(k)\right] = 0$$

Normal Equation Derivation

$\widehat{\theta}(k)$: Parameter estimate which solves

$$\frac{\partial V(\widehat{\theta}(k))}{\partial\widehat{\theta}(k)} = 0$$

Is given by:

$$\boxed{\Phi(k-1)\Phi(k-1)^T\,\widehat{\theta}(k) = \Phi(k-1)\,Y(k)}$$

Normal equation

Normal Equation Derivation

Normal equation:

$$\boxed{\Phi(k-1)\Phi(k-1)^T\,\widehat{\theta}(k) = \Phi(k-1)\,Y(k)}$$

where

$$\Phi(k-1)\Phi(k-1)^T = \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)$$

$$\Phi(k-1)Y(k) = \sum_{i=1}^{k} \phi(i-1)y(i)$$

Normal Equation Derivation

Normal equation:

$$\Phi(k-1)\Phi(k-1)^T\,\hat{\theta}(k) = \Phi(k-1)\,Y(k)$$

Is equivalent to

$$\boxed{\left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right]\hat{\theta}(k) = \sum_{i=1}^{k}\phi(i-1)\,y(i)}$$

Batch Least Squares Estimation

The solution of the normal equation

$$\left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right]\hat{\theta}(k) = \sum_{i=1}^{k}\phi(i-1)\,y(i)$$

Is given by:

$$\hat{\theta}(k) = \left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right]^{\#}\sum_{i=1}^{k}\phi(i-1)\,y(i)$$

Pseudo inverse

Pseudo Inverse

Let $A \in \mathcal{R}^{n\times m}$ and $A^\sharp$ be its pseudo-inverse

Then $A^\sharp$ has the dimension of A^T and satisfies:

- $A\,A^\sharp\,A = A$
- $A^\sharp\,A\,A^\sharp = A^\sharp$
- $A^\sharp A$ Is Hermitian

In this case, since $A = \Phi\Phi^T$

$\Phi = \left[\ \phi(0)\cdots\phi(k-1)\ \right]$

$$\boxed{A\,A^\sharp\,\Phi = \Phi}$$

Batch Least Squares Estimation

Assume that we have collected sufficient data and the data has sufficient richness so that

$$\left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right] = \phi(0)\phi^T(0) + \phi(1)\phi^T(1) + \cdots + \phi(k-1)\phi^T(k-1)$$

has rank n.

Then,

$$\hat{\theta}(k) = \left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right]^{-1}\sum_{i=1}^{k}\phi(i-1)\,y(i)$$

Recursive Least Squares (RLS)

Assume that we have collected k-1 sets of data and have computed $\hat{\theta}(k-1)$

$$\hat{\theta}(k-1) = \left[\sum_{i=1}^{k-1} \phi(i-1)\phi^T(i-1)\right]^{-1} \sum_{i=1}^{k-1} \phi(i-1)y(i)$$

Then, given a new set of data: $y(k) \quad \phi(k-1)$

We want to find $\hat{\theta}(k)$ in a recursive fashion:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + [\, correction \;\; term \,]$$

Recursive Least Squares Algorithm

Define the ***a-priori*** output estimate:

$$\hat{y}^o(k) = \phi^T(k-1)\hat{\theta}(k-1)$$

and the ***a-priori*** output estimation error:

$$e^o(k) = y(k) - \phi^T(k-1)\hat{\theta}(k-1)$$

The RLS algorithm is given by:

$$\boxed{\hat{\theta}(k) \;=\; \hat{\theta}(k-1) + F(k)\phi(k-1)\, e^o(k)}$$

Recursive Least Squares Gain

The RLS gain $F(k)$ is defined by

$$F^{-1}(k) \;=\; \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)$$

Therefore,

$$F^{-1}(k) \;=\; F^{-1}(k-1) + \phi(k-1)\phi^T(k-1)$$

Using the matrix inversion lemma, we obtain

$$\boxed{F(k) \;=\; F(k-1) - \frac{F(k-1)\,\phi(k-1)\,\phi(k-1)^T\,F(k-1)}{1+\phi(k-1)^T\,F(k-1)\,\phi(k-1)}}$$

Recursive Least Squares Derivation

Define the least squares gain matrix $F(k)$

$$\boxed{F(k) \;=\; \left[\sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)\right]^{-1}}$$

Therefore,

$$\hat{\theta}(k) = F(k) \sum_{i=1}^{k} \phi(i-1)y(i)$$

Recursive Least Squares Derivation

Notice that since,

$$
\begin{aligned}
F^{-1}(k) &= \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1) \\
&= \underbrace{\sum_{i=1}^{k-1} \phi(i-1)\phi^T(i-1)}_{F^{-1}(k-1)} + \phi(k-1)\phi^T(k-1)
\end{aligned}
$$

then,

$$F^{-1}(k) = F^{-1}(k-1) + \phi(k-1)\phi^T(k-1)$$

Recursive Least Squares Derivation

Notice that

$$\hat{\theta}(k) = F(k) \sum_{i=1}^{k} \phi(i-1)y(i)$$

Is equivalent to,

$$\hat{\theta}(k) = F(k)\,[\underbrace{\sum_{i=1}^{k-1} \phi(i-1)\,y(i)}_{F^{-1}(k-1)\hat{\theta}(k-1)} + \phi(k-1)\,y(k)]$$

and

$$F^{-1}(k-1) = F^{-1}(k) - \phi(k-1)\phi^T(k-1)$$

Recursive Least Squares Derivation

Therefore plugging the previous two results,

$$
\begin{aligned}
\hat{\theta}(k) = \; & F(k)\,\left[\left(F(k)^{-1} - \phi(k-1)\,\phi^T(k-1)\right)\hat{\theta}(k-1)\right. \\
& \left. + \phi(k-1)\,y(k)\right]
\end{aligned}
$$

And rearranging terms, we obtain

$$
\boxed{
\begin{aligned}
\hat{\theta}(k) = \; & \hat{\theta}(k-1) \\
& + F(k)\phi(k-1)\,\underbrace{\left[y(k) - \phi^T(k-1)\hat{\theta}(k-1)\right]}_{e^o(k)}
\end{aligned}
}
$$

Recursive Least Squares Estimation

Define the ***a-priori*** output estimate:

$$\hat{y}^o(k) = \phi^T(k-1)\hat{\theta}(k-1)$$

and the ***a-priori*** output estimation error:

$$e^o(k) = y(k) - \phi^T(k-1)\hat{\theta}(k-1)$$

The RLS algorithm is given by:

$$\boxed{\hat{\theta}(k) = \hat{\theta}(k-1) + F(k)\phi(k-1)\,e^o(k)}$$

Recursive Least Squares Estimation

Recursive computation of $F(k)$

$$F^{-1}(k) = \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)$$

Therefore,

$$F^{-1}(k) = F^{-1}(k-1) + \phi(k-1)\phi^T(k-1)$$

Using the matrix inversion lemma, we obtain

$$\boxed{F(k) = F(k-1) - \frac{F(k-1)\,\phi(k-1)\,\phi(k-1)^T F(k-1)}{1+\phi(k-1)^T F(k-1)\,\phi(k-1)}}$$

Recursive Least Squares Estimation

Matrix inversion lemma:

$$F^{-1}(k) = F^{-1}(k-1) + \phi(k-1)\phi^T(k-1)$$

- Multiply by $F(k-1)$ on the right and $F(k)$ on the left:

$$F(k-1) = F(k) + F(k)\,\phi(k-1)\,\phi(k-1)^T F(k-1)$$

- Multiply by $\phi(k-1)$ on the right:

$$F(k-1)\phi(k-1) = F(k)\phi(k-1) + F(k)\phi(k-1)\underbrace{\phi(k-1)^T F(k-1)\phi(k-1)}_{scalar}$$

Recursive Least Squares Estimation

Matrix inversion lemma:

- Rearranging terms,

$$F(k-1)\phi(k-1) = \left[1+\phi(k-1)^T F(k-1)\phi(k-1)\right] F(k)\phi(k-1)$$

- Solving for $F(k)\phi(k-1)$

$$F(k)\phi(k-1) = \frac{F(k-1)\phi(k-1)}{\left[1+\phi(k-1)^T F(k-1)\phi(k-1)\right]}$$

Recursive Least Squares Estimation

Matrix inversion lemma:

- Plug

$$F(k)\phi(k-1) = \frac{F(k-1)\phi(k-1)}{\left[1+\phi(k-1)^T F(k-1)\phi(k-1)\right]}$$

into

$$F(k) = F(k-1) - F(k)\,\phi(k-1)\,\phi(k-1)^T F(k-1)$$

to obtain

$$\boxed{F(k) = F(k-1) - \frac{F(k-1)\,\phi(k-1)\,\phi(k-1)^T F(k-1)}{1+\phi(k-1)^T F(k-1)\,\phi(k-1)}}$$

RLS Estimation Algorithm

$$e^o(k) = y(k) - \phi^T(k-1)\hat{\theta}(k-1)$$

$$F(k) = F(k-1) - \frac{F(k-1)\,\phi(k-1)\,\phi(k-1)^T\,F(k-1)}{1+\phi(k-1)^T\,F(k-1)\,\phi(k-1)}$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + F(k)\phi(k-1)\,e^o(k)$$

Initial conditions:

$$F(0) = F^T(0) > 0 \qquad \hat{\theta}(0)$$

RLS Estimation Algorithm

A-posteriori version:

$$e^o(k) = y(k) - \phi^T(k-1)\hat{\theta}(k-1)$$

$$e(k) = \frac{e^o(k)}{1+\phi^T(k-1)F(k-1)\phi(k-1)}$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + F(k-1)\phi(k-1)\,e(k)$$

$$F(k) = F(k-1) - \frac{F(k-1)\,\phi(k-1)\,\phi(k-1)^T\,F(k-1)}{1+\phi(k-1)^T\,F(k-1)\,\phi(k-1)}$$

RLS Estimation Algorithm

Define the parameter estimation error: $\boxed{\tilde{\theta}(k) = \theta - \hat{\theta}(k)}$

Notice that, since

$$y(k) = \phi^T(k-1)\theta$$

And the a-priori error is

$$e^o(k) = y(k) - \phi^T(k-1)\hat{\theta}(k-1)$$

We obtain,

$$\begin{aligned} e^o(k) &= \phi^T(k-1)\theta - \phi^T(k-1)\hat{\theta}(k-1) \\ &= \phi^T(k-1)\underbrace{\left[\theta - \hat{\theta}(k-1)\right]}_{\tilde{\theta}(k-1)} \end{aligned}$$

RLS Estimation Algorithm

Thus, the a-priori output estimation error can be written as

$$\boxed{e^o(k) = \phi^T(k-1)\tilde{\theta}(k-1)}$$

Similarly, define the **a-posteriori output and estimation error :**

$$\hat{y}(k) = \phi^T(k-1)\hat{\theta}(k)$$

$$e(k) = y(k) - \hat{y}(k)$$

then,

$$\boxed{e(k) = \phi^T(k-1)\tilde{\theta}(k)}$$

RLS Estimation Algorithm

Derivation of the RLS A-posteriori version:

$$\hat{\theta}(k) \;=\; \hat{\theta}(k-1) + F(k)\phi(k-1)\, e^{o}(k)$$

$$e^{o}(k) = y(k) - \phi^{T}(k-1)\hat{\theta}(k-1)$$

Remember that,

$$F(k)\phi(k-1) \;=\; \frac{F(k-1)\phi(k-1)}{\left[1 + \phi(k-1)^{T} F(k-1)\phi(k-1)\right]}$$

Thus,

$$\hat{\theta}(k+1) \;=\; \hat{\theta}(k) + \frac{F(k)\phi(k)}{1+\phi^{T}(k)F(k)\phi(k)}\, e^{o}(k+1)$$

RLS Estimation Algorithm

Multiplying by $\phi^{T}(k)$ to the left of

$$\tilde{\theta}(k+1) \;=\; \tilde{\theta}(k) - \frac{F(k)\phi(k)}{1+\phi^{T}(k)F(k)\phi(k)}\, e^{o}(k+1)$$

to obtain,

$$\underbrace{\phi^{T}(k)\tilde{\theta}(k+1)}_{e(k+1)} \;=\; \underbrace{\phi^{T}(k)\tilde{\theta}(k)}_{e^{o}(k+1)} - \frac{\phi^{T}(k)F(k)\phi(k)}{1+\phi^{T}(k)F(k)\phi(k)}\, e^{o}(k+1)$$

Thus,

$$\begin{aligned} e(k+1) \;&=\; e^{o}(k+1) - \frac{\phi^{T}(k)F(k)\phi(k)}{1+\phi^{T}(k)F(k)\phi(k)}\, e^{o}(k+1) \\ &=\; \frac{e^{o}(k+1)}{1+\phi^{T}(k)F(k)\phi(k)} \end{aligned}$$

RLS Estimation Algorithm

Therefore, from

$$\hat{\theta}(k+1) \;=\; \hat{\theta}(k) + F(k)\phi(k)\frac{e^{o}(k+1)}{1+\phi^{T}(k)F(k)\phi(k)}$$

We obtain,

$$\hat{\theta}(k+1) \;=\; \hat{\theta}(k) + F(k)\phi(k)\, e(k+1)$$

$$\boxed{e(k+1) \;=\; \frac{e^{o}(k+1)}{1+\phi^{T}(k)F(k)\phi(k)}}$$

RLS with forgetting factor

The inverse of the gain matrix in the RLS algorithm is given by:

$$F^{-1}(k) \;=\; F^{-1}(k-1) + \phi(k-1)\phi^{T}(k-1)$$

Its trace is given by:

$$\mathrm{tr}\left[F^{-1}(k)\right] \;=\; \mathrm{tr}\left[F^{-1}(k-1)\right] + |\phi(k-1)|^{2}$$

which always increases when $|\phi(k-1)| \neq 0$

RLS with forgetting factor

Similarly, the trace of the gain matrix is given by

$$\mathrm{tr}\,[F(k)] \;=\; \mathrm{tr}\,[F(k-1)] - \frac{|F(k-1)\phi(k-1)|^2}{1+\phi^T(k-1)F(k-1)\phi(k-1)}$$

always decreases when $|F(k-1)\phi(k-1)| \neq 0$

Problem: RLS eventually stops updating

RLS with forgetting factor

We can modify cost function to “forget” old data

$$V(\hat{\theta}(k)) = \frac{1}{2}\sum_{j=1}^{k} \lambda^{(k-j)} \left[y(j) - \phi^T(j-1)\,\hat{\theta}(k)\right]^2$$

$$0 < \lambda \leq 1$$

Key idea: Discount old data, e.g. the term

$$\lambda^{(k-1)} \left[y(1) - \phi^T(0)\,\hat{\theta}(k)\right]^2$$

is small when $\boldsymbol{k}$ is large since $\lim_{m\to\infty} \lambda^m = 0$

RLS with forgetting factor

$$\left.\begin{aligned} e^o(k+1) &= y(k+1) - \phi^T(k)\hat{\theta}(k) \\ e(k+1) &= \frac{e^o(k+1)}{1+\phi^T(k)F(k)\phi(k)} \\ \hat{\theta}(k+1) &= \hat{\theta}(k) + F(k)\phi(k)\,e(k+1) \end{aligned}\right\} \textit{same equations as RLS}$$

RLS gain with forgetting factor $0 < \lambda \leq 1$

$$F^{-1}(k+1) \;=\; \lambda\, F^{-1}(k) + \phi(k)\phi^T(k)$$

$$F(k+1) \;=\; \frac{1}{\lambda}\left[F(k) - \frac{F(k)\,\phi(k)\,\phi(k)^T\,F(k)}{\lambda + \phi(k)^T\,F(k)\,\phi(k)}\right]$$

RLS with forgetting factor

The gain of the RLS with FF may blow up

$$\mathrm{tr}\,[F(k)] \;=\; \frac{1}{\lambda}\mathrm{tr}\,[F(k-1)] - \frac{|F(k-1)\phi(k-1)|^2}{\lambda^2 + \lambda\phi^T(k-1)F(k-1)\phi(k-1)}$$

If $\phi(k)$ is not persistently exciting
(more on this later)

General PAA gain formula

$$F^{-1}(k+1) \;=\; \lambda_1(k)\, F^{-1}(k) + \lambda_2(k)\, \phi(k)\phi^T(k)$$

$$0 < \lambda_1(k) \leq 1 \qquad\qquad 0 \leq \lambda_2(k) < 2$$

- Constant adaptation gain: $\lambda_1(k) = 1,\ \lambda_2(k) = 0$
- RLS: $\lambda_1(k) = 1,\ \lambda_2(k) = 1$
- RLS with forgetting factor: $\lambda_1(k) < 1,\ \lambda_2(k) = 1$

General PAA gain formula

$$F^{-1}(k+1) \;=\; \lambda_1(k)\, F^{-1}(k) + \lambda_2(k)\, \phi(k)\phi^T(k)$$

$$0 < \lambda_1(k) \leq 1 \qquad\qquad 0 \leq \lambda_2(k) < 2$$

$$F(k+1) \;=\; \frac{1}{\lambda_1(k)} \left[F(k) - \frac{F(k)\,\phi(k)\,\phi(k)^T\,F(k)}{\frac{\lambda_1(k)}{\lambda_2(k)} + \phi(k)^T\,F(k)\,\phi(k)} \right]$$

$$F(0) = F^T(0) > 0$$