Recursive Least Squares

Konstantinos Galanos

In practice the estimation scheme is needed to be iterative, allowing the estimated model to be updated at each sample interval as new data become available.

By recalling the "*Batch Least Squares*" algorithm let us commence using equation:

$$\hat{\theta}(t) = \left(X^T(t)X(t)\right)^{-1} X^T(t)y(t) \tag{1}$$

where

$$X(t) = \begin{bmatrix} x^T(1) \\ x^T(2) \\ \vdots \\ x^T(t) \end{bmatrix}$$

$$y(t) = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(t) \end{bmatrix}$$

At time *t + 1* we obtain further measurements from the process which enables us to update the matrices above:

$$X(t+1) = \begin{bmatrix} x^T(1) \\ x^T(2) \\ \vdots \\ x^T(t) \\ \cdots \\ x^T(t+1) \end{bmatrix} = \begin{bmatrix} X(t) \\ \cdots \\ x^T(t+1) \end{bmatrix}$$

$$y(t+1) = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(t) \\ \cdots \\ y(t+1) \end{bmatrix} = \begin{bmatrix} y(t) \\ \cdots \\ y(t+1) \end{bmatrix} \tag{2}$$

The estimates at step *t + 1* are given by:

$$\hat{\theta}(t+1) = \left[X^T(t+1)X(t+1)\right]^{-1} X^T(t+1)y(t+1) \tag{3}$$

Let us consider the two terms on the right hand side of equation (3) one at a time.

$$X^T(t+1)X(t+1) = \left[X^T(t)x(t+1)\right]\begin{bmatrix} X(t) \\ \cdots \\ x^T(t+1) \end{bmatrix} = X^T(t)X(t) + x(t+1)x^T(t+1) \tag{4}$$

and

$$X^T(t+1)y(t+1) = \left[X^T(t)x(t+1)\right]\begin{bmatrix} y(t) \\ \cdots \\ y(t+1) \end{bmatrix} = X^T(t)y(t) + x(t+1)y(t+1) \tag{5}$$

Equations (4) and (5) give us the means to update equation (3) at every sample interval. However we do need to find a way to update the inverse of equation (4). Let us introduce some shorthand by denoting:

$$\left.\begin{aligned} P(t) &= \left[X^T(t)X(t)\right]^{-1} \\ B(t) &= X^T(t)y(t) \end{aligned}\right\} \tag{6}$$

Then we have:

$$\left.\begin{aligned} \hat{\theta}(t+1) &= P(t+1)B(t+1) \\ \hat{\theta}(t) &= P(t)B(t) \end{aligned}\right\} \tag{7}$$

Substituting equations (6) into equations (4) and (5) yields:

$$P^{-1}(t+1) = P^{-1}(t) + x(t+1)x^T(t+1) \tag{8}$$

and

$$B(t+1) = B(t) + x(t+1)y(t+1) \tag{9}$$

In order to get a direct update from *P(t)* to *P(t+1)* we need to apply the **Matrix Inversion Lemma**:

$$\left(A+BCD\right)^{-1}=A^{-1}-A^{-1}B\left(C^{-1}+DA^{-1}B\right)^{-1}DA^{-1} \tag{10}$$

By inspecting equations (8) and (10) we can set:

$$A=P^{-1}(t), \qquad B=x(t+1), \qquad C=1, \qquad D=x^{T}(t+1)$$

which gives a direct means of updating *P(t + 1)*:

$$P(t+1)=P(t)\left[I_m-x(t+1)\left(1+x^{T}(t+1)P(t)x(t+1)\right)^{-1}x^{T}(t+1)P(t)\right] \tag{11}$$

The only inversion in equation (11) is of the scalar term $1+x^{T}(t+1)P(t)x(t+1)$. Now let us define the error variable *ε(t + 1)*:

$$\varepsilon(t+1)=y(t+1)-x^{T}(t+1)\hat{\theta}(t) \tag{12}$$

Substituting equation (12) for *y(t + 1)* in equation (9) we get

$$B(t+1)=B(t)+x(t+1)x^{T}(t+1)\hat{\theta}(t)+x(t+1)\varepsilon(t+1) \tag{13}$$

Rearranging equations (7) we have

$$\left.\begin{aligned} B(t+1)&=P^{-1}(t+1)\hat{\theta}(t+1)\\ B(t)&=P^{-1}(t)\hat{\theta}(t)\end{aligned}\right\} \tag{14}$$

Substituting equations (14) into equation (13) for *B(t)* and *B(t + 1)* we get

$$\begin{aligned} &P^{-1}(t+1)\hat{\theta}(t+1)=P^{-1}(t)\hat{\theta}(t)+x(t+1)x^{T}(t+1)\hat{\theta}(t)+x(t+1)\varepsilon(t+1)\Rightarrow\\ &P^{-1}(t+1)\hat{\theta}(t+1)=P^{-1}(t+1)\hat{\theta}(t)+x(t+1)\varepsilon(t+1)\Rightarrow\\ &\hat{\theta}(t+1)=\hat{\theta}(t)+P(t+1)x(t+1)\varepsilon(t+1)\end{aligned} \tag{15}$$

Algorithms

Matrix Inversion Lemma RLS, version 1

At time step *t +1*

1. Form x(t + 1) using the new data.
2. Form $\varepsilon(t+1) = y(t+1) - x^T(t+1)\hat{\theta}(t)$
3. Form $P(t+1) = P(t)\left[I_m - \frac{x(t+1)x^T(t+1)P(t)}{1+x^T(t+1)P(t)x(t+1)}\right]$
4. Update $\hat{\theta}(t+1) = \hat{\theta}(t) + P(t+1)x(t+1)\varepsilon(t+1)$
5. Loop back to step (1).

Matrix Inversion Lemma RLS, version 2

At time step $t + 1$

1. Form x(t + 1) using the new data.
2. Form $\varepsilon(t+1) = y(t+1) - x^T(t+1)\hat{\theta}(t)$
3. Form $L(t+1) = \frac{P(t)x^T(t+1)}{1+x^T(t+1)P(t)x(t+1)}$
4. Update $\hat{\theta}(t)$ to obtain
   $\hat{\theta}(t+1) = \hat{\theta}(t) + L(t+1)\varepsilon(t+1)$
5. Form
   $P(t+1) = P(t) - L(t+1)\left[P(t)x(t+1)\right]^T$
6. Loop back to step (1).

Recursive Extended least squares

At time step $t + 1$

1. Form $\varphi(t+1)$ using new data $u(t+1), y(t+1)$ and
   $\varepsilon(t+1) = y(t+1) - \phi^T(t+1)\hat{\theta}(t)$
   where
   $\phi^T(t) = \left[y(t+1),\ldots,y(t-n_a),u(t-1),\ldots,u(t-n_b-1),\varepsilon(t-1),\ldots,\varepsilon(t-n_c)\right]$
2. $$P(t+1) = P(t)\left[I_m - \frac{\phi(t+1)\phi^T(t+1)P(t)}{1+\phi^T(t+1)P(t)\phi(t+1)}\right]$$
3. $$\hat{\theta}(t+1) = \hat{\theta}(t) + P(t+1)\phi(t+1)\varepsilon(t+1)$$

4. Loop back to step (1).

Approximate Maximum Likelihood

At time step *t + 1*

1. Form $\psi(t+1)$ using new data $u(t+1), y(t+1)$:
   $\psi^T(t) = \left[y(t-1),\ldots,y(t-n_a),u(t-1),\ldots,u(t-n_b-1),\eta(t-1),\ldots,\eta(t-n_c) \right]$
   and calculate the residual
   $\eta(t) = y(t) - \psi^T(t)\hat{\theta}(t)$
2. $$P(t+1) = P(t)\left[I_m - \frac{\psi(t+1)\psi^T(t+1)P(t)}{1+\psi^T(t+1)P(t)\psi(t+1)} \right]$$
3. $$\hat{\theta}(t+1) = \hat{\theta}(t) + P(t+1)\psi(t+1)\varepsilon(t+1)$$
   $$\eta(t) = \frac{\varepsilon(t)}{1+\psi^T(t)P(t-1)\psi(t)}$$
4. Loop back to step (1).

REFERENCES

1.
“Self-tuning Systems. Control and Signal Processing”, P.E. Wellstead and M.B. Zarrop, Wiley, ISBN 0471928836