A Simple Polynomial Estimator

Let $x(t)$ be a signal which is continuously observed in time starting at $t=0$. At time $T>0$, the signal is therefore known on the interval $[0, T]$. A simple technique for smoothing or predicting the signal is to approximate it by a polynomial on $[0, T]$ and use the polynomial as if it were the signal. In real-time operation this procedure requires the formalation of a new polynomial estimate at each time T. It is necessary to develop a scheme for the continuous updating of the approximating polynomial.

Suppose the signal $x(t)$ is approximated by the $(n-1)$th degree polynomial

$$P(T, t) = a_1(T) + a_2(T)t + \cdots + a_n(T)t^{n-1}$$

chosen to minimize

$$\int_0^T [x(t) - P(T, t)]^2 dt.$$

New coefficients $a_i(T)$ must be found for each T.

This simple estimation scheme is obviously only a special case of much more general schemes whose solution can be obtained in recursive form by solving a matrix Riccati differential equation.[1] The point of interest here, however, is that for the special scheme described above, the coefficients $a_i(T)$ satisfy an equation of the form

$$\dot{a}_i(T) = \frac{b_i}{T^i}\epsilon(T) \qquad i = 1, 2, \cdots, n$$

where $\epsilon(T)$ is the instantaneous error

$$\epsilon(T) = x(T) - P(T, T)$$

and the b_i's are constants.

Thus to implement the scheme, it is necessary only to store the constants b_i. It is not necessary to find or store a matrix Riccati solution.

The solution to this problem can be obtained as a simplification of the more general approach, but it is perhaps nicer to proceed directly.

Straightforward application of the projection theorem (or alternatively setting of partial derivatives to zero) yields the necessary conditions

$$\int_0^T [x(t) - a_1(T) - a_2(T)t - \cdots - a_n(T)t^{n-1}]t^k dt = 0$$

for $k=0, 1, 2, \cdots, n-1$, and for all T.

Differentiation with respect to T yields

$$\epsilon(T)T^k = \int_0^T [\dot{a}_1(T) + \dot{a}_2(T)t + \cdots + \dot{a}_n(T)t^{n-1}]t^k dt$$

or

$$\epsilon(T)T^k = \dot{a}_1(T)\frac{T^{k+1}}{k+1} + \dot{a}_2(T)\frac{T^{k+2}}{k+2} + \cdots + \dot{a}_n(T)\frac{T^{k+n}}{k+n}.$$

Manuscript received November 28, 1966. This research was supported by Joint Service Contract Nonr 225 (83).

[1] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *J. Basic Engrg., Trans. ASME*, ser. D 83, pp. 95–108, March 1961.

Dividing by T^k and writing these equations in matrix form, one obtains

$$\begin{bmatrix} 1 & 1/2 & 1/3 \cdots 1/n \\ 1/2 & 1/3 & 1/4 \cdots 1/(n+1) \\ 1/3 & & \\ \vdots & & \\ 1/n & \cdots & 1/(2n-1) \end{bmatrix} \begin{bmatrix} \dot{a}_1(T)T \\ \dot{a}_2(T)T^2 \\ \vdots \\ \dot{a}_n(T)T^n \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \epsilon(T).$$

Thus

$$\dot{a}_i(T) = \frac{b_i}{T^i}\epsilon(T)$$

where b_i is the ith component of the solution of the equation

$$Hb = 1$$

where 1 is the vector with unity in each component and H is the Hilbert matrix $[1/(i+j-1)]$.

Since the Hilbert matrix H is ill-conditioned for large n,[2] severe numerical errors may occur if the b vector is found by use of a standard numerical procedure. An alternat tive algorithm is given by Trench.[3]

Actually, an explicit (although perhaps not greatly practical) expression for the components b_i can be readily obtained. The Hilbert matrix

$$H = \left[\frac{1}{i+j-1}\right]$$

is a special case of the Cauchy matrix

$$C = \left[\frac{1}{\lambda_i + \mu_j}\right].$$

A general expression for the inverse of a Cauchy matrix is given by Trench and Scheinok[4] and this can be applied to the present case to obtain

$$b_j = \sum_{i=1}^{n} \frac{(-1)^{i+j}}{(i+j-1)} \frac{(n-1+i)!(n-1+j)!}{(n-i)!(n-j)!} \cdot \frac{1}{(i-1)!^2(j-1)!^2}.$$

As a final remark, it appears that the simple form of this estimator cannot be extended to similar situations. For example, a corresponding filter does not seem to exist for a discrete-time polynomial filter.

David G. Luenberger
Stanford University
Stanford, Calif.

[2] M. Newmann and J. Todd, "The evaluation of matrix inversion programs," *J. Soc. Indust. Appl. Math.*, vol. 6, pp. 466–476, December 1958.

[3] W. F. Trench, "An algorithm for the inversion of finite Hankel matrices," *J. Soc. Indust. Appl. Math.*, vol. 13, pp. 1102–1107, December 1965.

[4] W. F. Trench and P. A. Scheinok, "On the inversion of a Hilbert type matrix," *SIAM Review*, vol. 8, pp. 57–61, January 1966.