NASA Contractor Report 178381

LEAST SQUARES LINEAR LAGS
AND LIMITED MEMORY FILTERS

(NASA-CR-178381) LEAST SQUARES LINEAR LAGS AND LIMITED MEMORY FILTERS (Wagner (Daniel H.) Associates) 29 p Avail: NTIS HC A03/MF A01 CSCL 12A

N88-12339

Unclas
G3/65 0105441

Joseph H. Discenza

DANIEL H. WAGNER, ASSOCIATES, INC.
Hampton, Virginia

Purchase Order L-16074C
October 1987

NASA
National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

ABSTRACT

Pure autoregressive (AR) models which are linear in the short term, that is when a variable can be predicted by linear regression on a limited number of past observations, are discussed. When evenly-spaced observations are available, a fixed set of AR coefficients can be calculated independent of the data. For filtering purposes, such a lag structure can be implemented recursively with an efficient algorithm. The method of computing variance recursively is also derived. A complete algorithm is presented in the appendix.

INTRODUCTION

Many important problems require the prediction of a variable sequentially based only on previous observations on that variable. The classic autoregressive (AR) model is

$$y_k = \sum_{i=1}^{n} \alpha_i y_{k-i} + w_k \ , \tag{1}$$

w_k being a second moment ergodic white noise sequence (see for instance Groupe [1984] Chapter 8). Several models with a different α_i are well known. Irving Fisher (1) studied a distributed lag structure with lag length n, of the following form:

$$y_k^e = \frac{1}{\sum_{i=1}^{n} i} [n y_{k-1} + (n-1)\, y_{k-2} + \dots\dots + y_{k-n}]. \tag{2}$$

This yields lag coefficients α_i,

$$\alpha_i = \frac{n+1-i}{\sum_{i=1}^{n} i} = \frac{2(n+1-i)}{n(n+1)} \tag{3}$$

which insures that

$$\sum_{i=1}^{n} \alpha_i = 1 \quad . \tag{4}$$

The average lag (defined as the sum of the weighted time periods) is

$$\bar{\alpha} = \sum_{i=1}^{n} i\alpha_i = \frac{\sum_{i=1}^{n}(ni+i-i^2)}{\sum_{i=1}^{n} i}. \tag{5}$$

Then

$$\bar{\alpha} = n+1 - \frac{\sum_{i=1}^{n} i^2}{\sum_{i=1}^{n} i} = n+1 - \frac{2n+1}{3} = \frac{n+2}{3}. \tag{6}$$

Another commonly used expectations model is called naive or static expectations,

$$y_k^e = y_{k-1}, \tag{7}$$

which is the Fisher equation with n=1.

Other forms of expectations operators, which have been investigated principally with reference to price expectations, include extrapolative, adaptive, and various ad hoc distributed lags. Some empirical tests have been done: see Turnovsky (3) and Turnovsky and Wachter (4).

The Least Squares Operator

An alternative method of forming expectations is by defining a linear trend using a specific number of observations, i.e.:

$$y_k^e = a^e + b^e x_k + \varepsilon_k \tag{8}$$

where a^e and b^e are estimated from the data.

By choosing the x scale so that $x_k=0$ and the independent variables of the past observations are $x_{k-i} = -i$, the model becomes $y_k^e = a^e + b^e \cdot 0 = a^e$. We now show how to obtain the prediction for y_k *without actually estimating a or b*, as follows. The least squares estimates of a and b are:

$$b^e = \frac{n \Sigma x_i y_i - \Sigma x_i \Sigma y_i}{n \Sigma x_i^2 - (\Sigma x_i)^2} \tag{9}$$

and

$$a^e = \overline{y} - b^e \overline{x} . \tag{10}$$

Letting $D = n\Sigma x_i^2 - (\Sigma x_i)^2$ and substituting D into (9) and then into (10) yields

$$a^e = \frac{\Sigma y_i}{n} - \frac{n(\Sigma x_i y_i)\Sigma x_i}{nD} + \frac{(\Sigma x_i)^2 \Sigma y_i}{nD} . \tag{11}$$

Solving for The Lag Coefficients

Next, reorder and expand terms in (11) to obtain:

$$a^e = \frac{1}{n}y_1 + \frac{1}{n} \quad y_2 + \ldots\ldots\ldots\ldots + \frac{1}{n} y_n$$

$$+ \frac{(\Sigma x_i)^2}{nD} y_1 + \frac{(\Sigma x_i)^2}{nD} \quad y_2 \quad + \ldots\ldots\ldots\ldots + \frac{(\Sigma x_i)^2}{nD} y_n$$

$$- (-1) \frac{\Sigma x_i}{D} y_1 - (-2) \frac{\Sigma x_i}{D} \quad y_2 \; - \; \ldots\ldots\ldots\ldots \; -(-n) \; \frac{\Sigma x_i}{D} y_n. \tag{12}$$

Now we wish to obtain $a^e = \alpha_1 y_1 + \alpha_2 y_2 + \ldots + \alpha_n y_n$.

Summing the coefficients of y_j above, we obtain the required AR coefficents which are, of course, independent of y:

$$\alpha_j = \frac{1}{n} + \frac{\left(\sum_{i=1}^{n} -i\right)^2}{n\left[n \cdot \sum_{i=1}^{n}(-i)^2 - \left(\sum_{i=1}^{n}-i\right)^2\right]} + \frac{j \cdot \sum_{i=1}^{n}-i}{n \cdot \sum_{i=1}^{n}(-i)^2 - \left(\sum_{i=1}^{n}-i\right)^2} \quad . \tag{13}$$

Substituting the equations for sums of numbers and sums of squares into (13) yields a simplified expression for the AR coefficients:

$$\alpha_j = \frac{1}{n}\left[1 + \frac{3(n+1)}{(n-1)} - \frac{6j}{n(n-1)}\right] = \frac{2(2n+1-3j)}{n(n-1)} \quad . \tag{14}$$

The important feature of this model is that it provides an AR lag structure whose coefficients follow directly from the hypothesis of the limited memory linear least squares model, and depend only on the order of the model and not on the data. Table 1 provides the coefficents from (14) for lag lengths up to 8.

TABLE 1. Linear Least Squares Expectations Lag Coefficients Age of Observation (in Time Periods)

	1	2	3	4	5	6	7	8
2	2	-1						
3	4/3	1/3	-2/3					
4	1	1/2	0	-1/2				
5	8/10	5/10	2/10	-1/10	-4/10			
6	2/3	7/15	4/15	1/15	-2/15	-1/3		
7	4/7	3/7	2/7	1/7	0	-1/7	-2/7	
8	1/2	11/28	2/7	5/28	1/14	-1/28	-1/7	-1/14

The average lag of the least squares expectations operator is

$$\bar{\alpha} = \sum_{i=1}^{n} i\alpha_i = \sum_{i=1}^{n} \frac{2i\ (2n + 1 - 3i)}{n(n-1)}$$

$$= \frac{2(2n+1)}{n(n-1)} \sum_{i=1}^{n} i - \frac{6}{n(n-1)} \sum_{i=1}^{n} i^2$$

$$= \frac{2(2n+1) \bullet \quad n(+1)}{n(n-1) \bullet \quad 2} - \frac{6 \quad \bullet n(n+1)(2n+1)}{n(n-1) \quad \bullet \quad 6}$$

$$= 0. \tag{15}$$

Predictions for Different Time Periods

Coefficients generated by (14) provide least squares expectations for one observation after the last. Coefficients can also be obtained to predict y further in the future. Let m be the number of observations missing between the last observation and the prediction period. If the index of the required expectation is zero, then

$$b^e = \frac{(n-m)\ \Sigma x_i\, y_i - \Sigma x_i\, \Sigma x_i}{(n-m)\ \Sigma x_i{}^2\ - (\ \Sigma x_i\ )^2} \tag{16}$$

where all summations are from m+1 to n.

Then, let

$$D = (n-m) \sum_{i=m+1}^{n} x_i^2 - \left(\sum_{i=m+1}^{n} x_i \right)^2$$

$$= (n-m) \left(\sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{m} x_i^2 \right) - \left(\sum_{i=1}^{n} x_i - \sum_{i=1}^{m} x_i \right)^2 . \tag{17}$$

Substituting for summations using the appropriate equations yields

$$D = (n-m)^2\ (n-m-1)(n-m+1)/12. \tag{18}$$

With some manipulation, the expression for the lag coefficients for expectations m observations forward is then

$$\alpha_j = \frac{1}{n-m} + \frac{\{n(n+1) - m(m+1)\}^2}{4(n-m)D} - \frac{\{n(n+1) - m(m+1)\}\,(m+j)}{2D} \;. \qquad (19)$$

For example, if expectations for three periods in the future are needed, and a lag length of four is chosen, then n=6 and m=2. Using (13) yields

$$y_k^e = 1.6y_{k-3} + 0.7y_{k-4} - 0.2y_{k-5} - 1.1y_{k-6} \cdot$$

The sum of the lag weights is the same as when m=0,

$$\sum_{j=1}^{n-m} \alpha_j = \sum_{i=m+1}^{n} \left(\frac{1}{n-m} + \frac{\{n(n+1) - m(m+1)\}^2}{4(n-m)D} - \frac{\{n(n+1) - m(m+1)\}\,(m+i)}{2D}\right)$$

$$= \frac{n-m}{n-m} + \frac{(n-m)\{n(n+1) - m(m+1)\}^2}{4(n-m)D} - \frac{\{n(n+1) - m(m+1)\}^2}{4D}$$

$$= 1. \qquad (20)$$

The average lag of the future-period least squares expectations operator calculated as

$$\sum_{j=1}^{n-m} (m+j)\,\alpha_j$$

is also zero, but the proof is left to the reader. Table 2 shows coefficients for m=0 to 5 and order 2 to 5.

Table 2: Least Squares Expectations Coefficients Where m Periods Are Skipped Between the Last Observation and the Prediction.

Number of Lag Coefficients (n-m)

m	2		3			4				5			
0	2	-1	4/3	1/3	-2/3	1	.5	0	-.5	.8	.5	.2	-.1
1	3	-2	11/6	1/3	-7/6	1.3	.6	-.1	-.8	1	.6	.2	-.2
2	4	-3	7/3	1/3	-5/3	1.6	.7	-.2	-1.1	1.2	.7	.2	-.3
3	5	-4	17/6	1/3	-13/6	1.9	.8	-.3	-1.4	1.4	.8	.2	-.4
4	6	-5	10/3	1/3	-8/3	2.2	.9	-.4	-1.7	1.6	.9	.2	-.5

Filtering Applications

In this section we show how the lag operator is implemented recursively, for the case m=0.

First, find two successive estimates y_{k-1}^e and y_k^e:

$$y_{k-1}^e = \sum_{i=1}^{n} \alpha_i\, y_{k-1-i} = \alpha_n\, y_{k-1-n} + \sum_{i=1}^{n-1} \alpha_i\, y_{k-1-i} \tag{21}$$

and

$$y_k^e = \sum_{i=1}^{n} \alpha_i\, y_{k-i} = \sum_{i=0}^{n-1} \alpha_{i+1}\, y_{k-1-i}$$

$$= \alpha_1\, y_{k-1} + \sum_{i=1}^{n-1} \alpha_{i+1}\, y_{k-1-i}\,. \tag{22}$$

The difference between (21) and (22) is

$$y_k^e - y_{k-1}^e = \alpha_1\, y_{k-1} - \alpha_n\, y_{k-1-n} + \sum_{i=1}^{n-1} (\alpha_{i+1} - \alpha_i)\, y_{k-1-i}. \tag{23}$$

But we note that

$$\alpha_{i+1} - \alpha_i = 2\left[\frac{-3(i+1) + 3i}{n(n-1)}\right] = \frac{-6}{n(n-1)}\,. \tag{24}$$

Substituting (24) into (23) we obtain the recursive expression

$$y_k^e = y_{k-1}^e + \alpha_1\, y_{k-1} - \alpha_n\, y_{k-1-n} - \frac{6}{n(n-1)} \sum_{i=1}^{n-1} y_{k-1-i}\,. \tag{25}$$

The recursion is completed by noting that

$$\sum_{i=1}^{n-1} y_{k-1-i} = \sum_{i=1}^{n-1} y_{k-2-i} + y_{k-2} - y_{k-1-n} . \qquad (26)$$

The recursive approach requires that the n most recent observations be stored, but at each iteration only y_{k-1}, y_{k-2}, and y_{k-1-n} enter the calculations. Except for startup processing (the first n observations) the amount of processing for such a filter is independent of the lag period n. An algorithm is provided in the appendix.

Variance of the Prediction

We now derive the variance of the prediction, $\sigma^2 y_k^e$. This variance is determined as follows (from Kmenta, p 228).

$$\sigma^2_{y_k^e} = E\left[\left(y_k^e - E\,(y_k^e)\right)^2\right]$$

$$= E\,\{\left[(a^e + b^e x_k) - (a + bx_k)\right]^2\}$$

$$= E\,\{\,(a^e - a)^2\,\} + E\,\{\,(b^e - b)^2\, x_k^2\,\} + 2E\,\{(a^e - a)(b^e - b)\, x_k\}$$

$$= \mathrm{Var}\,(a^e) + x_k^2\, \mathrm{Var}(b^e) + 2x_k\, \mathrm{Cov}\,(a^e, b^e) \qquad (27)$$

Now we know from linear regression that, where $x'_i = x_i - \bar{x}$,

$$\mathrm{Var}\,(b^e) = \frac{\sigma^2}{\Sigma x_i'^2} \quad ,$$

$$\mathrm{Var}\,(a^e) = \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\Sigma x_i'^2}\right] ,$$

and

$$\mathrm{Cov}\,(a^e, b^e) = -\ \bar{x}\left[\frac{\sigma^2}{\Sigma x_i'^2}\right] , \tag{28}$$

where σ^2 is Var (ε_k) in (8).

Substituting, we obtain

$$\sigma^2 y_k^e = \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\Sigma x_i'^2}\right] + x_k^2\left[\frac{\sigma^2}{\Sigma x_i'^2}\right] - 2x_k\ \bar{x}\left[\frac{\sigma^2}{\Sigma x_i'^2}\right]$$

$$\sigma^2 y_k^e = \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\Sigma x_i'^2} + \frac{x_k^2}{\Sigma x_i'^2} - \frac{2x_k\ \bar{x}}{\Sigma x_x'^2}\right]$$

$$\sigma^2 y_k^e = \sigma^2\left[\frac{1}{n} + \frac{(x_k - \bar{x})^2}{\Sigma x_i'^2}\right] . \tag{29}$$

The expression in (29) gives the variance of the predicted *mean value* of y for a given x_k. Since the actual observed value of y varies about the true mean value with variance σ^2 (independent of the variance of y^e), the predicted value of an *individual* observation will still be given by y^e but will have variance (from Draper and Smith, p.24):

$$\sigma^2 + \sigma^2 y_k^e = \sigma^2 \left[1 + \frac{1}{n} + \frac{(x_k - \bar{x})^2}{\Sigma x_i'^2} \right], \tag{30}$$

and since $x_k = 0$,

$$(x_k - \bar{x})^2 = (- \bar{x})^2 = \frac{(n+1)^2}{4}. \tag{31}$$

So,

$$\sigma^2 y_k^e = \sigma^2 \left[\frac{1}{n} + \frac{(n+1)^2}{4 \sum_{i=1}^{n} x_i'^2} \right] \tag{32}$$

or,

$$\sigma^2 y_k^e = \sigma^2 \left[1 + \frac{1}{n} + \frac{(n+1)^2}{4 \sum_{i=1}^{n} x_i'^2} \right] . \tag{33}$$

Whether (32) or (33) is the appropriate equation will depend on the application. If the filter is predicting the mean value of y (such as the actual position of a target) then (32) should be used, because the variance can be made arbitrarily small by increasing the number of observations. Conversely, if we need to predict the next observation such as for certain search applications, the (33) is used and the minimum variance is σ^2 no matter how many observations are used. In the remaining analysis we will use (33), however the development using (32) is nearly identical.

Variance of the Disturbance

To estimate σ^2 we use s^2, an unbiased estimator where $y_i' = y_i - \bar{y}$:

$$s^2 = \frac{1}{n-2} \sum_{i=1}^{n} (y_i - a^e - b^e x_i)^2 .$$

$$= \frac{1}{n-2} \sum_{i=1}^{n} \left[y_i - (\bar{y}_i - b^e \bar{x}) - b^e x_i \right]^2$$

$$= \frac{1}{n-2} \sum_{i=1}^{n} (y_i' - b^e x_i)^2$$

$$= \frac{1}{n-2} \left[\sum_{i=1}^{n} (y_i')^2 - 2b^2 \sum_{i=1}^{n} x_i' y_i' + (b^e)^2 \sum_{i=1}^{n} x_i'^2 \right] .$$

But we know (see Kmenta, p. 208) that

$$b^e \sum_{i=1}^{n} x_i'^2 = \sum_{i=1}^{n} x_i' y_i'$$

and

$$(b^e)^2 \sum_{i=1}^{n} x_i'^2 = b^e \sum_{i=1}^{n} x_i' y_i' .$$

So we obtain,

$$s^2 = \frac{1}{n-2}\left[\sum_{i=1}^{n} y_i'^2 - b^e \sum_{i=1}^{n} x_i' y_i'\right] \tag{34}$$

which is a well-known expression for s^2. The limited-memory filter is a unique model where the x_i are known, therefore, some simplification is possible. First, express

$$b^e = \frac{\Sigma x_i' y_i'}{\Sigma x_i'^2}$$

and subsitute it into (34) to yield,

$$s^2 = \frac{1}{n-2}\left[\sum_{i=1}^{n} y_i'^2 - \frac{\left(\sum_{i=1}^{n} x_i' y_i'\right)^2}{\Sigma x_i'^2}\right]. \tag{35}$$

Now, it is easy to show that

$$\sum_{i=1}^{n} y_{k-i}' = \Sigma y_{k-i}^2 - n \bar{y}_k^2 \tag{36}$$

which yields

$$s^2 = \frac{1}{n-2}\left[\Sigma y_{k-i}^2 - \frac{1}{n}\left(\sum_{i=1}^{n} y_{k-i}\right)^2 - \frac{\left(\sum_{i=1}^{n} x_i' y_i'\right)^2}{\sum_{i=1}^{n} x_i'^2} \right] . \qquad (37)$$

For an efficient recursive calculation of s^2, we do not wish to calculate the summations in (37) each time. Of course, since n is constant and $x_{k-i} = -i$,

$$\sum_{i=1}^{n} x_i'^2 = \frac{n(n+1)(2n+1)}{6} - \frac{n(n+1)^2}{4} . \qquad (38)$$

Furthermore,

$$\sum_{i=1}^{n} y_{k-i} = \sum_{i=1}^{n} y_{k-1-i} - y_{k-1-n} + y_{k-1} \qquad (39)$$

and

$$\sum_{i=1}^{n} y_{k-i}^2 = \sum_{i=1}^{n} y_{k-1-i}^2 - y_{k-1-n}^2 + y_{k-1}^2 . \qquad (40)$$

Now, to compute the final summation term recursively, we write

$$\sum_{i=1}^{n} x_i' y_i' = \sum_{i=1}^{n} \left(-i + \frac{n+1}{2}\right)\left(y_{k-i} - \bar{y}\right)$$

$$= \sum_{i=1}^{n} \left(-i + \frac{n+1}{2}\right) y_{k-i} - \bar{y} \sum_{i=1}^{n} \left(-i + \frac{n+1}{2}\right)$$

$$\sum_{i=1}^{n} x_i' \, y_i' = \sum_{i=1}^{n} -i \; y_{k-i} + \frac{n+1}{2} \sum_{i=1}^{n} y_{k-i} \,. \qquad (41)$$

Now, we expand the first tem in (41), as follows, for epochs k and k-1:

$$- \sum_{i=1}^{n} i \; y_{k-i} = -y_{k-1} - 2y_{k-2} - \ldots - ny_{k-n}$$

$$- \sum_{i=1}^{n} i \; y_{k-1-i} = - y_{k-2} - \ldots - (n-1)y_{k-n} - ny_{k-1-n} \cdot$$

Then we subtract, yielding

$$- \sum_{i=1}^{n} i \; y_{k-i} + \sum_{i=1}^{n} i \; y_{k-1-i} = - \sum_{i=1}^{n} y_{k-i} + ny_{k-1-n} \cdot \qquad (42)$$

Now we write the expression for the cross product term at epoch k and epoch k-1, using (41)

$$\left(\sum_{i=1}^{n} x_i' \, y_i' \right)_k = - \sum_{i=1}^{n} i \; y_{k-i} + \frac{n+1}{2} \sum_{i=1}^{n} y_{k-i} \cdot$$

and

$$\left(\sum_{i=1}^{n} x_i' \, y_i'\right)_{k-1} = -\sum_{i=1}^{n} i \, y_{k-1-i} + \frac{n+1}{2}\sum_{i=1}^{n} y_{k-1-i} \, .$$

Subtracting the last two equations and substituting (39) and (42), we obtain finally,

$$\left(\sum_{i=1}^{n} x_i' \, y_i'\right)_{k} - \left(\sum_{i=1}^{n} x_i' \, y_i'\right)_{k-1} = -\sum_{i=1}^{n} y_{k-i} + n y_{k-1-n} + \frac{n+1}{2}\left(y_{k-1} - y_{k-1-n}\right)$$

(43)

which is the last required difference equation.

The algorithm which implements this recursive model is described in the appendix.

Statistical Justification for the Model

Under what circumstance is such a linear model valid? We can think of many real world situations where processes are linear (or can be transformed to ones that are) over the short term but in the long term may be very non-linear. One with which we are familiar is a linear process whose slope is subject to randomly occurring jumps. Under this assumption, we have the true model,

$$y_i = a + bx_i + \varepsilon_i, \; i \geq k\text{-}n \tag{44}$$

where k-n is the epoch of the most recent jump. Of course, n is the unknown but we can estimate its value. This estimate may give us two kinds of problems in the estimation of a and b. First, if we choose n too large, and we include data points which are not part of the true model, our regression is *biased*. However, until the next jump occurs, we are at least *consistent*. On the other hand, if we choose n too small, then we are omitting usable data from the regression, which is *inefficient*.

We suggest that it might be possible, given a particular application, to choose n so as to minimize the expected total (bias and inefficiency) error.

Computational Results

Table 3 shows a simulated data set and the results of using the model with order n=8. The true model is $y_i = 18+2_i$, i=1,...,18 and $y_i = 54\text{-}i$, i=19,...,50. The observation error introduced in the simulation is normal with $\sigma=3$.

Figure 1 shows the behavior of the model for n=6, 8, 10, and 15, respectively. Note that the calculated variance (using equation (32)) increases for a time (about n observations) after the jump at observation 18, an indication that the filter in some sense "detects" the jump.

We should note that much more powerful filters are available to deal with the type of data discussed herein, such as Kalman Filters with short- and long-term processes, but all such models require considerably more processing and are much more general. The filter presented here is narrowly-defined but extremely easy to compute.

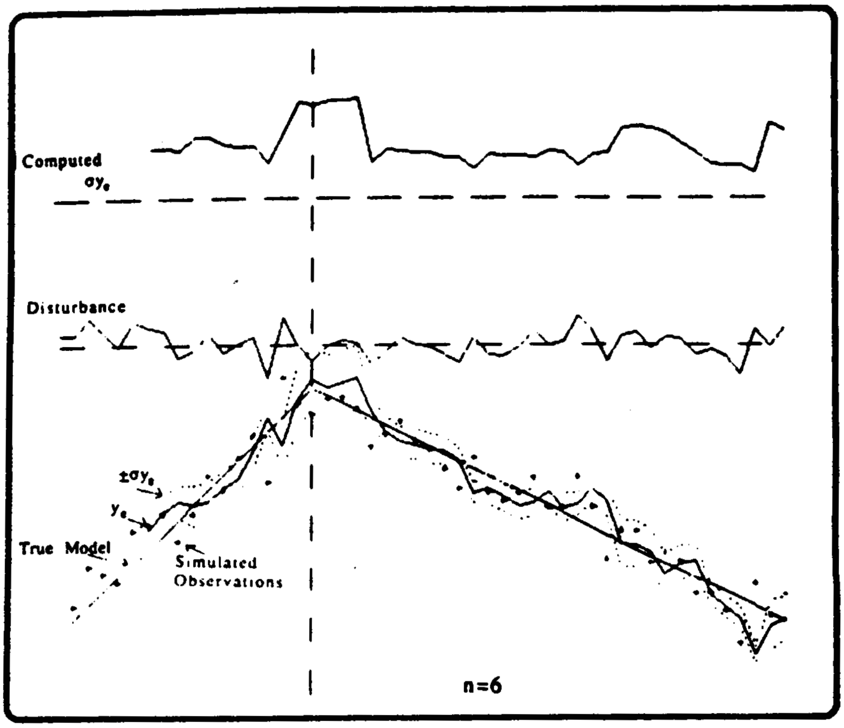
<u>Table 1</u>

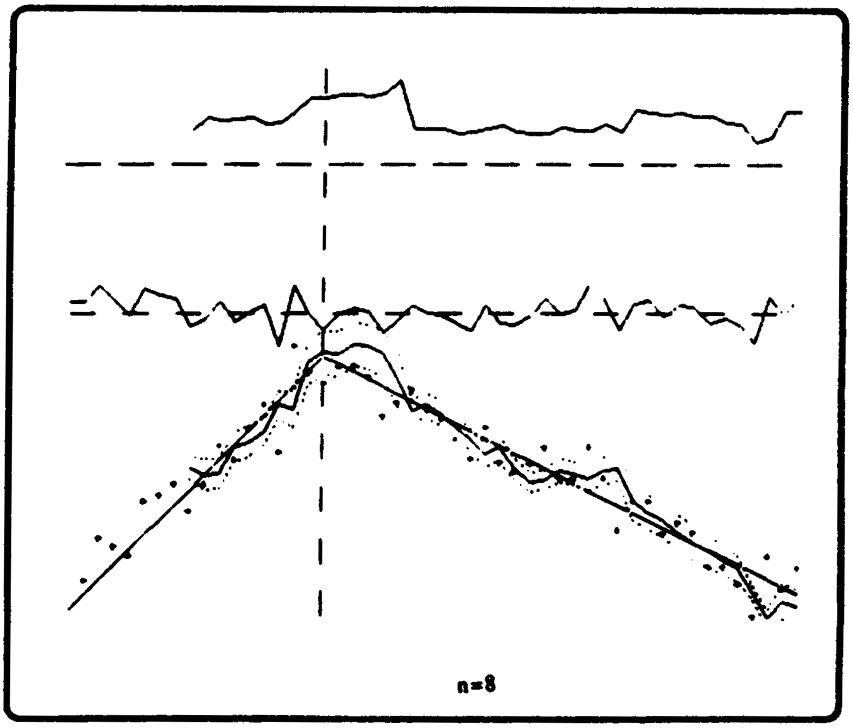
Simulation Results

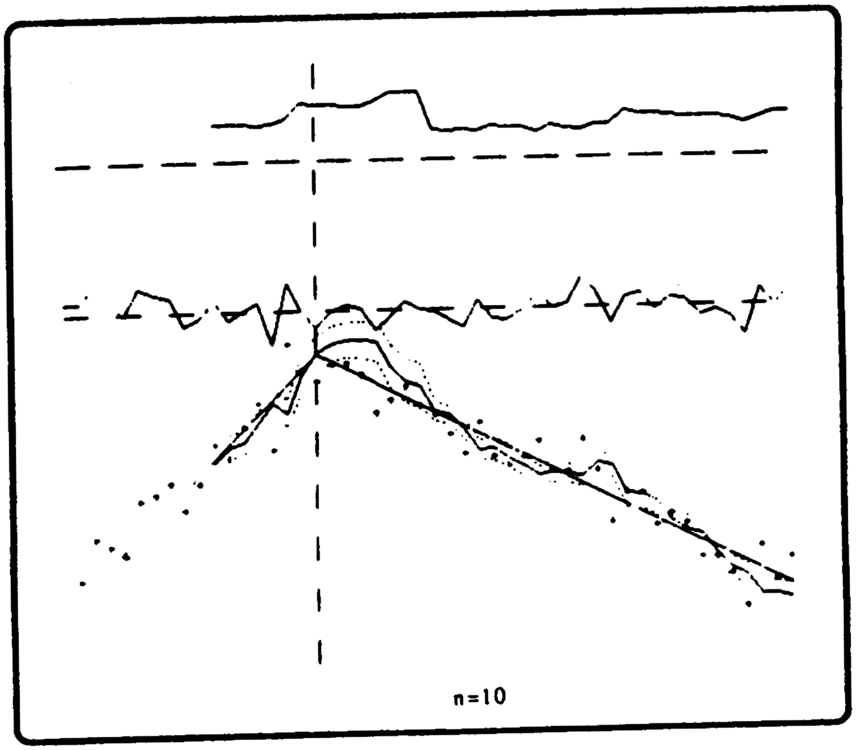
Number	Observ	Predict	Error	Variance	Sigma
1	22.3367				
2	24.0347				
3	29.6256				
4	28.5831				
5	27.4105				
6	34.6796				
7	35.4258				
8	36.9346				
9	33.2763	39.1537	5.8773	4.009	2.434
10	36.8247	38.1588	1.3341	4.358	2.088
11	42.0522	38.3724	-3.6797	3.792	1.947
12	40.1222	41.8262	1.7040	4.200	2.049
13	44.3455	42.6113	-1.7343	4.368	2.090
14	47.5987	43.8529	-3.7458	3.195	1.787
15	41.1860	47.5027	6.3166	3.498	1.870
16	55.5352	46.8285	-8.7067	5.834	2.415
17	51.9135	53.2506	1.3371	9.065	3.011
18	50.4021	54.5623	4.1602	8.911	2.985
19	52.7145	54.3888	1.6743	9.730	3.119
20	52.8151	55.7838	2.9687	9.683	3.112
21	51.2435	55.3974	4.1539	9.419	3.069
22	45.9523	54.3735	8.4212	10.218	3.197
23	47.6973	51.1269	3.4296	14.498	3.808
24	49.4900	46.6388	-2.8512	2.465	1.570
25	46.8479	47.4743	.6264	2.598	1.612
26	45.5741	46.3428	.7688	2.556	1.599
27	42.9773	44.5775	1.6002	1.986	1.409
28	40.2959	42.8569	2.5610	2.147	1.465
29	44.4757	40.8748	-3.6009	2.474	1.573
30	39.7585	41.7626	2.0041	2.991	1.730
31	38.6393	39.1273	.4880	2.134	1.461
32	39.6311	37.2192	-2.4119	1.872	1.368
33	41.7540	37.4178	-4.3363	1.888	1.374
34	37.8070	38.7905	.9835	2.721	1.649
35	37.6862	38.2805	.5944	2.382	1.543
36	41.9468	38.6149	-4.3318	2.383	1.544
37	37.6959	38.6889	.9929	3.434	1.853
38	30.6637	38.9533	8.2896	2.105	1.451
39	34.6891	34.7433	.0542	6.192	2.488
40	34.5748	33.1207	-1.4540	5.326	2.308
41	30.1492	32.2185	2.0692	5.110	2.260
42	31.6565	30.4034	-1.2531	5.333	2.309
43	30.3123	29.1756	-1.1367	4.945	2.224
44	25.7946	27.8916	2.0970	4.680	2.163
45	25.7283	26.6604	.9321	3.552	1.885
46	23.4342	25.6085	2.1743	3.241	1.800
47	18.8451	22.1803	3.3352	1.086	1.042
48	27.1559	18.6698	-8.4860	1.590	1.261
49	22.6779	20.9714	-1.7065	5.798	2.408
50	25.5353	20.2484	-5.2869	5.918	2.433

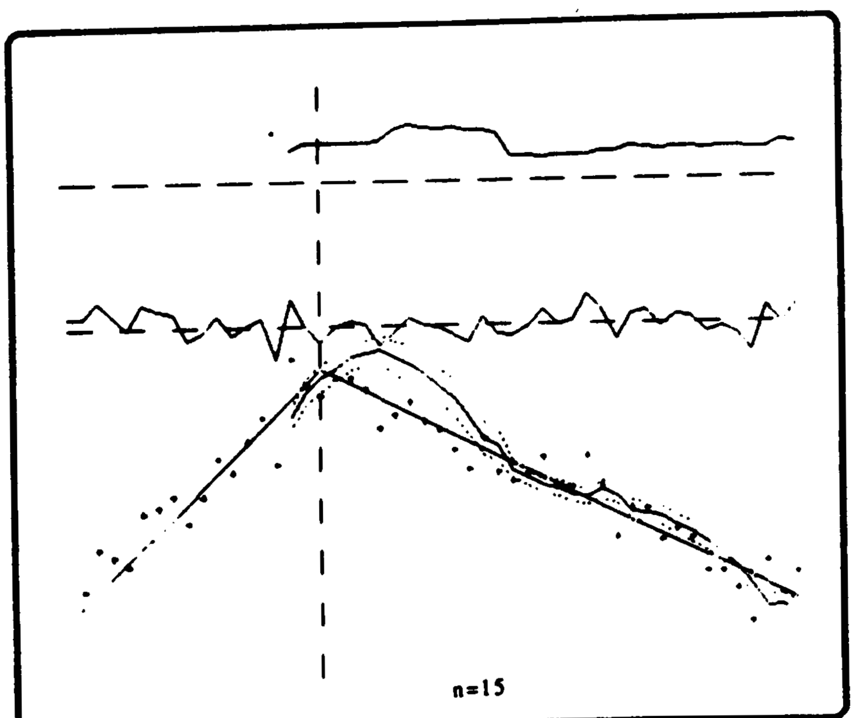
FIGURE 1

Results of Simulated Data Using Limited-Memory Filter






n=10


n=15

APPENDIX

SEQUENTIAL ALGORITHM FOR Y^e

BEGIN

Select n; then, $C_0 = \frac{6}{n(n-1)}$

and

$$\alpha_i = \frac{2(n+1-3i)}{n(n-1)}, \; i=1,\ldots,n \; .$$

Get n observations.

Compute $$y^e_{n+1} = \sum_{i=1}^{n} \alpha_i y_{n+1-i}$$

and

$$\sum_{i=1}^{n-1} y_{n-i} \; .$$

Set $k=n+1$.

REPEAT

Observe y_k.

INC (k,1)

Compute $$\sum_{i=1}^{n-1} y_{k-1-i} = \left(\sum_{i=1}^{n-1} y_{k-2-i}\right) + y_{k-2} - y_{k-1-n} \; .$$

Compute $$y^e_k = y^e_{k-1} + a_1 y_{k-1} - a_n y_{k-1-n} - C_0\left(\sum_{i=1}^{n-1} y_{k-1-i}\right).$$

END

RECURSIVE ALGORITHM FOR VAR (Y^e)

BEGIN

Compute $C_1 = \Sigma x'^2 = \frac{n(n+1)(2n+1)}{6} - \frac{n(n+1)^2}{4}$

and $C_2 = 1 + \frac{1}{n} + \frac{(n+1)^2}{4 \cdot C_1}$ or $\frac{1}{n} + \frac{(n+1)^2}{4C_1}$.

Get n observations.

Compute $\sum_{i=1}^{n} y_{n+1-i}$,

$\sum_{i=1}^{n} y_{n+1-i}^2$,

$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_{n+1-i}$,

$\Sigma xy'_{n+1} = \sum_{i=1}^{n} \left(x_{n+1-i} - \bar{x} \right) \left(y_{n+1-i} - \bar{y} \right)$,

and

$$\sigma_{n+1}^2 = C_0 \cdot \left[\frac{1}{n-2} \left\{ \sum_{i=1}^{n} y_{n+1-i}^2 - \frac{1}{n} \left(\sum_{i=1}^{n} y_{n+1-i} \right)^2 - \frac{\left(\Sigma x'y'_{n+1} \right)^2}{C_1} \right\} \right]$$

Set k = n+1 .

REPEAT

Observe y_k .

INC (k,1) .

Compute $$\sum_{i=1}^{n} y_{k-i} = \sum_{i=1}^{n} y_{k-1-i} + y_{k-1} - y_{k-1-n} ,$$

$$\Sigma x'y'_k = \Sigma x'y'_{k-1} - \sum_{i=1}^{n} y_{k-i} + n y_{k-1-n} + \frac{n+1}{2}\left(y_{k-1} - y_{k-1-n}\right) ,$$

$$\sum_{i=1}^{n} y_{k-i}^2 = \sum_{i=1}^{n} y_{k-1-i}^2 + (y_{k-1})^2 - (y_{k-1-n})^2 ,$$

and

$$\sigma^2 = C_2 \cdot \left[\frac{1}{n-2}\left\{\sum_{i=1}^{n} y_{k-i}^2 - \frac{1}{n}\left(\sum_{i=1}^{n} y_{k-i}\right)^2 - \frac{\left(\Sigma x'y'_k\right)^2}{C_1}\right\}\right] .$$

END

REFERENCES

(1) Fisher, Irving, The Theory of Interest, NY (1930), (1954).

(2) Kmenta, J., Elements of Econometrics, New York: MacMillan (1971).

(3) Turnovsky, S.J., "Empirical Evidence on the Formation of Price Expectations", Journal of the American Statistical Association 65, (1970): 1441-54.

(4) Turnovsky, S.J., and Wachter, M.L., "A Test of the 'Expectations Hypothesis' Using Directly Observed Wage and Price Expectations", Review of Economic Statistics 54 (1972): 47-54.

(5) Groupe, Identification and Adaptive Filtering, Malobar, FL, Krieger (1984).

(6) Draper, N. and Smith, H., Applied Regression Analysis, New York, Wiley (1966).

National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-178381		

4. Title and Subtitle	5. Report Date
Least Squares Linear Lags and Limited Memory Filters	October 1987
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Joseph H. Discenza	
	10. Work Unit No.
	505-63-91-02
9. Performing Organization Name and Address	11. Contract or Grant No.
Daniel H. Wagner, Associates, Inc. Hampton Office Suite 301, 27 West Queens Way Hampton, VA 23669	L-16074C
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
National Aeronautics and Space Administration Langley Research Center Hampton, VA 23665-5225	14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor: David Chestnutt

16. Abstract

Pure autoregressive (AR) models which are linear in the short term, that is when a variable can be predicted by linear regression on a limited number of past observations, are discussed. When evenly-spaced observations are available, a fixed set of AR coefficients can be calculated independent of the data. For filtering purposes, such a lag structure can be implemented recursively with an efficient algorithm. The method of computing variance recursively is also derived. A complete algorithm is presented in the appendix.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Statistics Least squares lags Limited memory filters Autoregressive prediction model	Unclassified - Unlimited Subject Category 65

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	28	A03