NOTE

A METHOD FOR ANALYZING EXPONENTIAL DECAYS*

PAUL MARSHALL

Department of Chemical Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, U.S.A.

(Received 23 October 1986)

Abstract—The non-linear least-squares routine KPSI is presented for the analysis of data from counting experiments which follow single exponential decays. It gives appropriate weighting to individual points, is compact (52 statements) and converges more rapidly than the widely used CURFIT program of Bevington. The routine quantifies the goodness of fit and the uncertainty in the decay coefficient. KPSI is particularly useful for microcomputer analysis when interfaced directly to laboratory equipment. A FORTRAN listing is included which may easily be adapted for BASIC.

INTRODUCTION

Many "real time" experiments rely on detection of signals which fall exponentially with time. Examples include kinetic studies in which resonance fluorescence, laser-induced fluorescence or chemiluminescence is used as the diagnostic of reactant concentration, measurements of excited-state lifetimes and studies involving radio-isotopes. If the signals are weak single pulse counting methods can be employed, with the counts accumulated in a multichannel scaler, often linked to a microcomputer for storage and display of the data. After capture the data may be transferred to a larger computer for analysis, but clearly it would be convenient to interpret decays in the course of the experiments. A short program KPSI which uses little memory and therefore is particularly suited to small and inexpensive microcomputers, is described for the analysis of a signal \( y \), which decays as a function of time \( t \). \( y \) is fitted to the form \( f(t) \) where

\[
f(t) = A \exp(-kt) + B
\]

where \( k \) is the pseudo first-order decay coefficient and \( A \) and \( B \) are constants.

A number of ways of extracting \( k \), especially where the background \( B \) is zero, have been discussed elsewhere (Wentworth, 1965; Swain et al., 1980; Demas, 1983; Hovanec & Ward, 1984). Here it is simply noted that subtracting a measurement made after a "long" time \( (y_n) \) to eliminate the \( B \) term requires certainty that the first term has completely decayed, gives excessive weight to \( y_n \) and may introduce errors if there is a second slow process, on a much longer timescale than \( 1/k \), which affects \( y \). Non-linear least-squares programs have been published which fit data to the form of equation (1) such as LSG and LSKIN1 (DeTar, 1968) and CURFIT (Bevington, 1969), which are relatively lengthy and do not always converge for poor data (Swain et al., 1980). KORE (Swain et al., 1980) is conveniently short, yields the uncertainty in \( k \) and quantifies the goodness of fit.

The routine KPSI given here is a modification of KORE which takes proper weighting of multichannel scaler data into account and uses an alternative statistical analysis to derive the goodness of fit and the SD of \( k \), whilst retaining KORE's compactness. KPSI is a subroutine able to be included in a data collection program interfaced to experimental apparatus.

PRINCIPLES

The method of least-squares aims to minimize

\[
\chi^2 = \sum_{i=1}^{n} \frac{(y_i - f(t_i))^2}{w_i}
\]

with respect to the parameters of \( f(t) \), where \( w_i \) represents a statistical weighting factor equal to the reciprocal of the variance of \( y \) (Bevington, 1969). The importance of correct weighting, particularly for poor data, has been emphasized previously (Demas, 1983). For pulse counting experiments Poisson statistics apply so that \( w_i = 1/y_i \). Minimizing equation (2) for \( A \) and \( B \) and solving the simultaneous equations in terms of \( k \) yields optimized values of \( A \) and \( B \) for any particular \( k \):

\[
A = \left\{ \sum y_i^{-1} \sum \exp(-kt_i) n \frac{\exp(-kt) y_i^{-1}}{\sum \exp(-kt) y_i^{-1}} \right\} \left\{ \sum y_i^{-1} \sum \exp(-2kt_i y_i^{-1}) n \right\}^{-1}
\]

\[
B = \left\{ \sum y_i^{-1} \sum \exp(-kt_i) y_i^{-1} \right\} \left\{ \sum \exp(-kt) y_i^{-1} \right\}^{-1}
\]

* Work supported by the U.S. Department of Energy, Office of Basic Energy Sciences, under grant DE-FG02-84ER13224.
SUBROUTINE KPSI(NPTS, K, A, B, Y, TBASE, SDK, RCS)
DIMENSION Y(256),E(0:256)
REAL K,M
DATA E(0~F.DELTA2.S2,RCS,A12,A13/1.,4.,0.,0.,0.,0.,0./
DO 102 I=1,NPTS
S2=S2+1./Y(I)
102 S1=0.
S2=0.
S4=0.
X=EXP(K)
DO 104 I=1,NPTS
E(I)=E(I-1)/X
104 E(I)=E(I-1)/X
Sl=Sl+E(I)/Y(I)
S3=S3+E(I)*E(I)/Y(I)
S4=S4+E(I)
A=(S4-S1*NPTS/S2)/(S3-Sl*Sl/S2)
B=(NPTS-A*Sl)/SZ
Tl=0.
T2=0.
DO 105 I=1,NPTS
U=I*E(I)/Y(I)
105 U=I*E(I)/Y(I)
Tl=Tl+(Y(I)-A*E(I)-B)*I+U
T2=T2+(2*A*E(I)-Y(I)+B)*I+U
IF (F.EQ.O) GOT0 106
DELTA=Tl/T2
K-K-(1.+F)*DELTA
M=1.
T=DELTA/DELTA
IF (T LE 0.7) THEN
M=.5
ELSE IF (T.LT.4) THEN
M=2.
END IF
IF (F*M.GT.0.5) THEN
F=F*M
ELSE
F=0.5
END IF
IF DELTA2=DELTA
IF (ABS(DELTA/K).LT.1E-5) F=O.
END IF
DO 107 I=1,NPTS
T=Y(I)-A*E(I)-B
RCS=RCS+T*T/Y(I)
107 RCS=RCS+T*T/Y(I)
A12=A12+(T-A*E(I))*U
A13=A13+(T-A*E(I))*U
SDK=SQRT((S3*52-Sl*Sl)/nET)/TBASE

Fig. 1. FORTRAN-77 Listing of KPSI.

\[ B = \left( n - A \sum \exp(-kt)y_i^{-1} \right) \left( \sum y_i^{-1} \right) \] (4)

where \( \Sigma \) means summation of the \( n \) data points over \( i \). \( k \) is sought such that \( g(k) = \partial y_i / \partial k \) is zero. An iterative technique is applied so that

\[ k_{i+1} = k_i - (1 + F)g(k_i)(\partial g(k_i) / \partial k)^{-1} \] (5)

gives an improved estimate \( k_{i+1} \) from an earlier estimate \( k_i \). The condition \( F = 0 \) corresponds to the traditional Newton-Raphson method but Swain et al. showed that provided \( F \) is adjusted suitably, a dramatically faster convergence can be obtained using \( F > 0 \), which forms their over-relaxation technique (Swain et al., 1980). Iteration is repeated with updated values of \( A \) and \( B \) until changes in \( k \) are sufficiently small.

Swain et al. used a correlation coefficient to quantify the goodness of fit, but a more convenient quantity which is widely tabulated is the reduced chi square \( \chi^2 \). The number of degrees of freedom \( \nu \) is \( n - 4 \) (Demas, 1983) and \( \chi^2 \) is easily calculated via equation (2) since \( \chi^2 = \chi^2/(n - 4) \): it should be approximately unity if equation (1) is indeed a fair fit to the observed data. Distributions of \( \chi^2 \) give bounds at various confidence levels and degrees of freedom. To obtain the uncertainty in \( k \) the error matrix \( e \) is employed (Bevington, 1969). With parameters \( \alpha \) and \( \beta \), the elements \( \sigma_{\alpha\beta} \) of the curvature matrix \( e \) are defined by \( \sigma_{\alpha\beta} = 0.5\partial^2 \chi^2/\partial \alpha \partial \beta \). The error matrix is

### Table 1. Comparison of KPSI and CURFIT with "perfect" data (entries rounded to 4 decimal places)

<table>
<thead>
<tr>
<th>Initial ( k )</th>
<th>Final ( k ), KPSI</th>
<th>Final ( k ), CURFIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.02</td>
<td>0.2000</td>
<td>0.0644</td>
</tr>
<tr>
<td>0.15</td>
<td>0.2000</td>
<td>0.1954</td>
</tr>
<tr>
<td>0.21</td>
<td>0.2000</td>
<td>0.1999</td>
</tr>
<tr>
<td>0.25</td>
<td>0.2000</td>
<td>0.1951</td>
</tr>
<tr>
<td>0.30</td>
<td>0.2000</td>
<td>0.1772</td>
</tr>
<tr>
<td>0.40</td>
<td>0.2000</td>
<td>0.1796</td>
</tr>
</tbody>
</table>
Table 2. Comparison of KPSI and CURFIT with scattered data* (entries rounded to 4 significant figures)

<table>
<thead>
<tr>
<th>Run</th>
<th>KPSI</th>
<th>CURFIT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>36.00</td>
<td>24.00</td>
</tr>
<tr>
<td>2</td>
<td>40.62</td>
<td>23.06</td>
</tr>
<tr>
<td>3</td>
<td>40.62</td>
<td>23.06</td>
</tr>
<tr>
<td>4</td>
<td>40.62</td>
<td>23.06</td>
</tr>
<tr>
<td>5</td>
<td>40.62</td>
<td>23.06</td>
</tr>
</tbody>
</table>

* 16 points: 60, 58, 42, 55, 44, 36, 44, 34, 32, 40, 30, 36, 38, 24, 33, 24.
† Execution time (s).

The variances of the parameters then lie on the leading diagonal of \( \epsilon \), thus \( \sigma_k = \epsilon_{11} \).

ALGORITHM

Figure 1 is a listing of the KPSI routine as used with a Digital Research FORTRAN-77 compiler for an IBM PC. On entry the datapoints are in the array \( Y \), NPTS being the number of points and TBASE the time per channel. K should contain an initial estimate of the decay coefficient (see later) and a rough value can be obtained from the expression

\[
K = \frac{3 \cdot \text{NPTS}}{\text{NPTS}} \log \left( \frac{Y(1)}{Y(\text{NPTS})} \right) - \frac{Y(\text{NPTS})}{3 \cdot Y(\text{NPTS})}. \tag{6}
\]

Operation is with single precision arithmetic and iteration stops when \( K \) changes by < 1 part in \( 10^5 \). This may be unnecessarily precise for most kinetic data but serves for demonstration. In the program the array of consecutive exponentials \( E \) is generated without repeated calls to the EXP function, which can save significant time for large numbers of points (Demas, 1983). The accumulated error after 256 points using this procedure is about 1 part in \( 10^5 \) (or < 2 parts in \( 10^6 \) with IBM interpreted Advanced BASIC, BASICA). The variables \( S1-S4 \) accumulate the summations of equations (3) and (4), \( F \) is the over-relaxation factor (Swain et al., 1980) and lines from 106 onwards evaluate the curvature matrix and calculate \( \sigma_k \). Values returned are the decay coefficient \( K \), its standard deviation SDK, the other fitted parameters \( A \) and \( B \), and \( \chi^2 \) as RCS. The array \( Y \) is unaltered.

The routine will fail if the initial value of \( K \) is so large that the Newton-Raphson correction overshoots and makes \( K \) negative. Like any comparable program KPSI will also fail if any of the data are zero since no meaningful weight can be attached for a Poisson distribution.

RESULTS

In Table 1 KPSI is compared to CURFIT, one of the more commonly referenced general least-squares programs (Bevington, 1969), using synthetic input data where \( A = 50, B = 20 \) and \( K = 0.2 \) for 16 points (TBASE = 1, \( i = 1-16 \)). KPSI has only 52 statements whereas CURFIT and its subroutines occupy 157. KPSI needs only an initial estimate of \( K \) while CURFIT requires estimates of all three parameters: here \( A \) and \( B \) were set to their "true" values to begin with. Table 1 illustrates the results. Clearly CURFIT is much more sensitive than KPSI to the starting value of \( K \). The crude estimate equation (6) is therefore suitable for KPSI which converges to the same answer irrespective of the starting value. Equation (6) could be added to the start of the subroutine.

Table 2 is a comparison of the two routines with a scattered dataset of 16 numbers. The first and last were used as initial estimates of \( A + B \) and \( B \) (not required for KPSI). Although CURFIT executes faster (30 s vs 60 s for KPSI) several consecutive runs, using the previous output as initial estimates, are needed to ensure convergence. KPSI converges in a single run and so has the shorter overall time.

The KPSI program is easily translated into other popular microcomputer languages such as BASIC. In fact it executes some three times more rapidly under BASICA on an IBM PC and for large numbers of points (more typical of an experimental dataset) the execution time is < 1 s per point, the exact time depending on their scatter. The algorithm presented here is therefore a useful addition to a computerized data acquisition system because of its small size, few input parameters and rapid convergence.

Acknowledgement—Thanks are due to Dr A. Fontijn for pointing out the need for this work and for his encouragement and advice.

REFERENCES