SplMod

(Version 3; June 1988)

Users Manual

Data Analysis Group, EMBL, Heidelberg
Technical Report EMBL-DA09 (June 1988)

Users manual for SplMod — A portable Fortran IV program
for analyzing sums of one-parameter functions,
with an option to approximate the model function by cubic B-splines.

Robert H. Vogel
sp@S-provencher.COM
http://S-provencher.COM

This manual was typeset with *D. E. Knuth*'s text formatting system TEX.
[Knuth, D.E., *The TEXbook,* (Addison-Wesley Publishing Company, Reading, Massachusetts, 1984)]

TABLE OF UPDATES

0.	MAY 84	Original version.
1.	APR 86	Error in call to `USEROU` corrected; a few minor changes.
		Error in call to `USEREX` corrected (argument `RESPON` was missing); subroutine name in `USERSI` and `USEREX` (for improved diagnostics) provided.
2.	JUN 88	Error in `INPUT` corrected (Control Variable `LINEPG` was misspelt); change in `OUTITR` to prevent formatting errors; error in `VARIAN` corrected (wrong argument was used for `SQRT` in `DP` version only).
		Error in `OUTCOR` corrected (Fortran logical unit number was incorrect in two `WRITE` statements).

TABLE OF CONTENTS

JUN 88

PAGE UPDATE SUMMARY

All pages have stayed the same as in the original version, MAY 84, and the 1st update, APR 86, except for slight changes in the wording for more complete description and better understanding.

1. PREFACE AND OVERVIEW

1.1 Distribution of SPLMOD

SPLMOD [1] is a program for analyzing sums of one-parameter functions (e.g., convoluted exponentials, as well as pure exponentials). Every attempt has been made to make the program easily transportable by adhering it (with one popular exception, mentioned in Sec 2.3) to *1966 ANSI Fortran IV*. Distribution of SPLMOD is unrestricted as long as it is free of charge and accompanied by a copy of this Users Manual.

We also ask that you distribute an unmodified copy (or at least note the changes that you have made) and that references be made in any publications using SPLMOD.

If you receive a second-hand copy of SPLMOD, send me a photocopy of the *Table of Updates* in your Users Manual and the date in the first line of SPLMOD (line `MAIN0001`). In this way, any outdated versions can be replaced and your name can be put on a mailing list for possible updates, etc. It would also be much safer if you let us send you a magnetic tape (Sec 2.1) containing the latest version of SPLMOD that you can copy and return; this would remove any worries about altered versions. We advise you to first make a file copy of the entire tape and to keep this unchanged. You should also inform me of address changes so that the mailing list can be updated.

1.2 Purpose of SPLMOD and Related Programs

SPLMOD is a general program for analyzing data represented by

$$y(t) \approx \sum_{j=1}^{N_\lambda} \alpha_j f(\lambda_j, t) + \sum_{\mu=1}^{N_\gamma} \gamma_\mu g_\mu(t) \tag{1}$$

where N_γ and the functional forms of $f(\lambda_j, t)$ and $g_\mu(t)$ are known and α_j, λ_j, γ_μ, and N_λ are to be estimated. Therefore exponentials, as well as convoluted exponentials, and other functions can be analyzed. Data of this type occur in numerous areas, including relaxation and tracer kinetics and radioactive and fluorescence decay.

In the case of exponentials it is sometimes necessary [2,3] to analyze the data as an integral over a continuous distribution (rather than a discrete sum), e.g., when solving Laplace's integral equation. Then CONTIN [4], a general purpose program for solving linear integral equations of the first kind and systems of (possibly ill-conditioned) linear algebraic equations, can be used.

There is an additional program, DISCRETE [5], for analyzing data as a discrete sum of exponentials plus a constant background only. To date, over 200 copies of DISCRETE are distributed all over the world and the program has proved to be extremely reliable and satisfying for most of the users.

SPLMOD was designed to offer all the possibilities of DISCRETE, but is no longer restricted to exponential models only, in addition, it is able to analyze up to N_D different datasets in parallel, each of which having the same nonlinear parameters λ but different amplitudes α and different γ. The main advantage of SPLMOD is its ability to approximate the model function by cubic B-splines. In doing a reliable and thorough analysis of your data, this will save you computer time in the orders of magnitude. The name of the program, SPLine MODel, is, in fact, derived from this approximation.

1.3 Design of SPLMOD

SPLMOD has been designed to be very flexible but still easy to use. (Since it can generate its own simulated data, we use it as a research tool, as well as for data analysis.) SPLMOD has been modelled after CONTIN and much of the structure and notation is the same. There are 39 *Control Variables* that are set to default values. You need only input the ones that you wish to change. You can also reset the default values. You need only read about the Control Variables corresponding to options that you intend to use. Similarly, of the more than 50 subprograms, there are 9 short `USER` subprograms (whose names all begin with the four characters `USER`) that can be used without change by most users, but you also can easily change any of these to suit your special needs.

1.4 Design of this Users Manual

1.4.1 What you must read

This Users Manual has also been designed to be easy to use. It has been closely modelled after the CONTIN manual [20] and the plan and much of the terminology and wording are identical. Sections 1 and 3 are essential reading for everyone. If SPLMOD has already been installed on your computer, then you can skip Sec 2. If you are only interested in *Applications*, then, after reading Secs 1 and 3, you can jump to the description of the Applications in Sec 6, where you will be referred to any parts of Sec 4 that are necessary reading. I have attempted to make the subsections in Sec 4 self-contained enough to be read separately. Very few users will have to read all of Sec 4. However, you should at least read the *Table of Contents* to become aware of all the possibilities of SPLMOD.

Read Section 5 only if you have difficulties. Section 6 provides useful examples of how you can utilize the possibilities of SPLMOD, and scanning it may give you new ideas on how to solve your own problems with SPLMOD. Section 7 is the list of references. This is far from complete and is mainly to provide examples of applications of SPLMOD. However some of the publications referenced in Sec 7 have useful bibliographies.

Section 8 contains useful summaries and indexes of important equations, Control Variables, and other terms. To save space, each term is not cross-referenced every time it appears in the text. Therefore, when you see a term for the first time, you should simply use Sec 8 to find out where it is defined.

I have attempted to write this manual so that only a minimal knowledge of mathematics and computers is required for its use, even at the risk of using very elementary terminology and notation. A major part involves heuristic discussions how to interpret results from SPLMOD.

1.4.2 If questions or problems arise

The great flexibility of SPLMOD could have two consequences:

(1) It may not be apparent to you how to best set up the problem or what combination of options is best. If such questions arise, please contact me.

(2) We will probably never be able to test SPLMOD with all possible combinations of options and obviously not with all possible `USER` subprograms. Difficulties may arise for certain combinations, either due to your improper

usage or to errors in SPLMOD. If you still have not resolved the difficulties after trying the suggestions in Sec 5, then send me the information specified in Sec 5.2. This will help us maintain SPLMOD.

1.4.3 Conventions and Notation

Fortran variables (e.g., `NY`), terms and phrases output by SPLMOD (e.g., `CORRELATION COEFFICIENTS`) are printed in all capitals with a teletype font in this manual.

Some terms with special meanings in this manual are capitalized and slanted (e.g., *Applications*) the first time they appear. The first time a Control Variable is defined, it is set into a box, and the location of the definitions of many terms can be found in Sec 8.

The term *Page* refers exclusively to the hand-numbered pages of the printed output from test runs in Sec 3.6.1. All the other references to locations in this manual are to section and subsection, not page.

Eq (3.2–1) refers to the equation numbered (1) in Sec 3.2. The term *Eq (1)* refers to the equation numbered (1) in the same section that the term *Eq (1)* occurs.

[2] refers to the reference numbered 2 in Sec 7.

In SPLMOD itself, comment cards set off with `C` in column 1 and `*` in columns 2–72 are messages of possible interest to you. In the `USER` subprograms, all comments are for you. The rest of the comments are mainly for me. They will only give you a rough idea of what is going on; e.g., arguments of the subprograms are usually not redefined in the comments of every subprogram. It is only intended that you are able to optionally modify the main, `BLOCK DATA`, and `USER` subprograms.

1.5 Acknowledgements

SPLMOD has been under development and testing since early 1980. I thank the users at EMBL and elsewhere; their problems stimulated many extensions and added options. I want to thank S.W. Provencher who developed the theory for the method and whose program packages DISCRETE and CONTIN provided a model for SPLMOD and this manual. I also thank J. Glöckner for his expert and patient work in typesetting this manual.

2. INSTALLING SPLMOD ON YOUR COMPUTER

2.1 Reading the Magnetic Tape

2.1.1 EMBL Tape format

All EMBL tapes are nine-track, unlabeled, 1600-bits-per-inch with 80 characters per logical record (i.e., Fortran card images) and 6 records (480 characters) per block. The tape reels are marked to show whether the tapes are in *ASCII* or *EBCDIC* code. When requesting a tape, you should specify one of these codes. Do not send us a blank tape of your own. We have written a large number of tapes (once and for all) and we can send you one that you can copy and return.

2.1.2 EMBL Tape contents

The files are ordered on the tape as follows:

File 1	SPLMOD Version `3SP`
File 2	" " `3DP`
File 3	Test data for SPLMOD (for both versions)
etc.	

A table of tape contents will be sent with each tape. You always need File 3 and either File 1 or File 2 (see Sec 2.2 to decide which).

It is a good idea to make a copy of the entire tape. Versions `3SP` and `3DP` are identical except that comment cards have been changed to Fortran statements and vice versa and the markers `!SP` and `!DP` have been inserted and deleted. Thus the pair of files can be used to check each other in case copying errors are suspected. More important, you may decide later to use the other version.

2.1.3 File contents

File 3 is listed in Sec 3.3.1. As mentioned in Sec 2.1.2, File 1 contains the same subprograms and the same number of cards in each subprogram as File 2 does. The 53 subprograms are ordered in each of the files as follows: main subprogram, `BLOCK DATA`, 9 `USER` subprograms, and the other 42 subprograms. They are written as Fortran card images and are labeled in columns 73–80. Columns 73–77 contain the name of the subprogram. (If the actual name has six characters, the third is deleted). Columns 78–80 consecutively number the *cards* in each subprogram; so the number on the last card is the number of cards in that subprogram. The last card of each subprogram has the characters `END` in columns 7–9 and blanks in columns 1–6 and 10–72. This card only comes at

the end of a subprogram. The first card in each subprogram also has a unique string that makes it easy to separate the subprograms: it has the characters `C++++` in columns 1–5. There are no separator cards between subprograms; the last card of one subprogram is followed immediately by the first card of the next.

The first card also has `VERSION 2DP (JUN 1988)` (or with `DP` replaced by `SP`); if your File 1 or File 2 has first cards with a date other than `JUN 1988`, then either your copy of SPLMOD or this Users Manual is outdated. Send me the date on your first cards and a photocopy of the Table of Updates at the front of this manual.

2.2 Version 3SP or 3DP?

I have attempted to use numerically stable algorithms [19]. Nevertheless, the required arithmetic precision increases if you have not properly scaled your problem. This will be more precisely discussed in Sec 3.6.4.3, but it is usually difficult to say *a priori* that these situations will never occur. Therefore, the single-precision Version `3SP` is only recommended for general use on machines (e.g., *CDC 7600*) that have single-precision words (i.e., Fortran `REAL` variables) of 60 bits or more; it will usually be safe on machines with 48-bit single precision too. Otherwise, Version `3DP`, which has key parts in double precision, should be used.

2.3 Possible necessary changes to SPLMOD

I believe that SPLMOD is in *1966 ANSI Standard Fortran IV* with the exception that some dummy arrays have a `1` in their rightmost `DIMENSION` specification. This is a popular exception even with libraries that are intended to be portable. Therefore, it is intended that SPLMOD can be run with no changes, except for the machine-dependent changes listed below and the problem-dependent changes described in Sec 3.5. This has been the case with test runs on the *DEC 2060* at *Max-Planck-Institut für Kernphysik* in Heidelberg and on the *IBM 3081* at *Universitäts-Rechenzentrum* in Heidelberg using their *Fortran77* and `WATFIV` compilers, as well as on the *VAX 8650* at EMBL.

2.3.1 Set the four machine-dependent parameters

These must be set (once and for all) in the `DATA` statement in Line `BLCKD039` (in Version `3SP`) or Line `BLCKD040` (in Version `3DP`) in the `BLOCK DATA` subprogram.

`RANGE` should be about two or three orders of magnitude smaller than the largest number (call it `BIG` so that `BIG` does not overflow and `1/BIG` does not underflow.
`RANGE` and `BIG` are `REAL` for Version `3SP` and `DOUBLE PRECISION` for Version `3DP`.

`SRANGE` is the same as `RANGE` except that `SRANGE` (and `BIG`) is in single precision (i.e., in Version `3SP`, `SRANGE=RANGE`).

`NIN` is the unit number for your input, i.e., all your data will be input using statements starting with `READ (NIN,...` .

`NOUT` is the unit number for your output, i.e., all your results will be output using statements starting with: `WRITE (NOUT,...` .

2.3.2 Underflow

In many parts of SPLMOD, underflows are expected to occur and be replaced by zero, which is the normal action of most compilers. However, if you happen to have a compiler that stops execution at underflows or prints a diagnostic at every underflow, no matter how many, then you will have to prevent this, e.g., by using a compilation option or switch or calling the `WATFIV` subprogram:

```
CALL TRAPS (0,0,99999,0,0)
```

Subprogram `USERTR` illustrates how you can avoid underflow in the library function `EXP`; this is actually necessary with a few compilers.

2.3.3 WATFIV

`WATFIV` was useful in testing SPLMOD, but is extremely slow in execution and not recommended for general usage. If you nevertheless want to test SPLMOD with `WATFIV`, you probably will have to make further changes, e.g., the `CALL TRAPS` in Sec 2.3.2.

2.3.4 Other Possible changes

There may be other required changes peculiar to your system. For example, you may be required to open your own input, output, and (only if you input `IUNIT` > 0) scratch files. Instructions on where to do this are given in the main subprogram in Lines `MAIN0152`–`MAIN0162`.

Other possible problem-dependent changes are discussed elsewhere: adjusting `DIMENSION` specifications (Sec 3.5), the default values of the Control Variables (Sec 3.4), or the `USER` subprograms (Sec 4.1). However, after making the changes discussed here in Sec 2.3, you should be ready to compile and run the test data from File 3.

2.4 Run with the Test Data

Compilation on the *VAX 8650*, *DEC 2060*, and with the `WATFIV` compiler on the *IBM 3081* produced no diagnostics, except for 260 `WATFIV` extensions due to a `1` in the rightmost `DIMENSION` specification of dummy arrays. Compilation on the *IBM 3081* with the *Fortran77* compiler produced 41 warnings due to the difference between *Fortran IV* and *Fortran77* in treating the Hollerith data type. *Fortran77* has the `CHARACTER` data type whereas in *Fortran IV* this type of data is identified under the guise of a name of one of the other types (we use `INTEGER` throughout SPLMOD).

The test run using Version `3DP` with the test data in File 3 took about 18 CPU seconds (not counting compilation and loading) on the *VAX 8650*. File 3 is shown in Sec 3.3.2.

The output from the test run on the *VAX 8650* with File 3 is shown in Sec 3.6.1. However, before you compare your output too closely with it, you should read Sec 3. Section 3.6 explains the output in detail and points out which output is strongly machine-dependent.

3. ESSENTIAL INFORMATION ON USING SPLMOD

3.1 Outline of Problems that SPLMOD Can Handle

SPLMOD analyzes data that can be represented (or approximated) by a sum of one-parameter functions:

$$y_k \approx \sum_{j=1}^{N_\lambda} \alpha_j f(\lambda_j, t_k) + \sum_{\mu=1}^{N_\gamma} \gamma_\mu g_\mu(t_k) =: \hat{y}(t_k) , \qquad k = 1, \ldots, N_y \tag{1}$$

where the data y_k generally contain experimental noise (and hence the $\approx$), the functional form of $f(\lambda_j, t_k)$ is known, and α_j, λ_j, and N_λ are to be estimated. The optional sum over the N_γ known $g_\mu(t_k)$ and unknown γ_μ permits, for example, an additional constant term, γ_1, to be included by setting $N_\gamma = 1$ and $g_1(t) = 1$. There are many optional Control Variables and `USER` subprograms, that allow you, for example, to analyze N_D different data sets simultaneously. (See Sec 4.1.2 for more details.)

Options to statistically weight the data in the least-squares analysis of Eq (1) are described in Sec 4.1.3.4. Options to generate simulated data with pseudo-random noise are described in Sec 4.1.7.2.

There are many other options and `USER` subprograms described in Secs 3.4 and 4.1. You can get an idea of the possibilities by scanning the Table of Contents.

3.2 Outline of Methods Used

I will only give a brief outline to enable you to use SPLMOD; see [1] and the references therein for more complete discussions.

From a statistical point of view, the best estimates of the α_j, λ_j, and γ_μ, for a given N_λ can be obtained by utilizing *Gauß*'s principle of least squares and minimizing

$$\mathtt{VAR} = \sum_{k=1}^{N_y} w_k (y_k - \hat{y}_k)^2 = \text{minimum} \tag{1}$$

where w_k are optional weights that you can assign and $\hat{y}_k$ is the model function defined in Eq (3.1–1).

Though SPLMOD has the option to analyze the data in terms of Eq (3.1–1) as it stands, it usually approximates the $f(\lambda_j, t_k)$, using cubic B-splines as shown in [1]:

$$f(\lambda, t_k) = \sum_{l=1}^{N_B} \beta_{lk} B_l(\lambda) \tag{2}$$

where $B(\lambda)$ is a normalized cubic B-spline [13] defined on an equally spaced knot sequence $z_1 = H(\lambda_{\min}), \ldots, z_{N_B} = H(\lambda_{\max})$ and $H(\cdot)$ is a `USER` supplied function, e.g., logarithm (See Sec 4.1.3.1 for a description of H = `USERTR`).

These B-spline approximations are good within experimental errors in the $y(t)$, and have the following advantages:

- It is easy to get derivatives of $f(\lambda, t)$ w.r.t. λ.
- The dimension of the least squares problem is reduced from N_y to $N_\lambda + N_\gamma$.
- The evaluation of the B-splines is easy and fast because of the equally spaced knot sequence.

3.3 Input Data Deck

3.3.1 Test data deck — File 3

Below you will find a listing of File 3 from the magnetic tape with the test input data. The line numbers have been added for convenience; they are not part of File 3 or the input data. Most of the input data cards contain a blank in column 1 and have a character in column 2.

An input data deck can consist of an arbitrary number of Data Sets. Each Data Set specifies a separate complete analysis that is to be performed. For example, Lines 1–74 and 75–140 in File 3 comprise two Data Sets. Running several Data Sets together in one run can be convenient because (as will be shown in Sec 3.4.1) some data from preceding Data Sets is preserved and need not be input again in later Data Sets.

```
TEST DATA SET 1      ***  CONVOLUTION  ***
NNL                        3.
NGAM                       1.
IWT                        2.
DOUSIN                     1.
IUSER     1                2.
LAST                      -1.
END
NINTT     1
NSTEND  160                1.            160.
  0.580954E+03  0.987631E+03  0.194205E+04  0.329751E+04  0.472440E+04
  0.600509E+04  0.705631E+04  0.809584E+04  0.893946E+04  0.944329E+04
  0.972025E+04  0.100739E+05  0.990493E+04  0.997284E+04  0.978328E+04
  0.971406E+04  0.956667E+04  0.930556E+04  0.909049E+04  0.865068E+04
  0.839509E+04  0.813499E+04  0.805614E+04  0.767957E+04  0.737493E+04
  0.720880E+04  0.702243E+04  0.683888E+04  0.654531E+04  0.628211E+04
  0.611013E+04  0.590699E+04  0.575305E+04  0.556679E+04  0.537367E+04
  0.525530E+04  0.514227E+04  0.501866E+04  0.493654E+04  0.480799E+04
  0.466767E+04  0.446559E+04  0.430753E+04  0.429396E+04  0.436461E+04
  0.414842E+04  0.401151E+04  0.397279E+04  0.389141E+04  0.380125E+04
  0.367661E+04  0.357186E+04  0.359123E+04  0.339443E+04  0.331909E+04
  0.345877E+04  0.322456E+04  0.317825E+04  0.315583E+04  0.303296E+04
  0.305516E+04  0.297074E+04  0.280511E+04  0.286016E+04  0.281720E+04
  0.271919E+04  0.268062E+04  0.267764E+04  0.266993E+04  0.255919E+04
  0.238792E+04  0.255743E+04  0.241663E+04  0.234405E+04  0.226293E+04
  0.228579E+04  0.222075E+04  0.212548E+04  0.218068E+04  0.209437E+04
  0.210139E+04  0.201835E+04  0.198056E+04  0.193850E+04  0.199847E+04
  0.185521E+04  0.193964E+04  0.192697E+04  0.183710E+04  0.181483E+04
  0.176789E+04  0.182283E+04  0.173610E+04  0.171855E+04  0.171283E+04
  0.159648E+04  0.163133E+04  0.157267E+04  0.158856E+04  0.152747E+04
  0.148819E+04  0.149406E+04  0.137873E+04  0.146651E+04  0.151674E+04
  0.142373E+04  0.133983E+04  0.139672E+04  0.140669E+04  0.133749E+04
  0.127427E+04  0.131390E+04  0.126248E+04  0.127558E+04  0.125996E+04
  0.121056E+04  0.127877E+04  0.119586E+04  0.123859E+04  0.107492E+04
  0.118758E+04  0.109693E+04  0.115358E+04  0.103315E+04  0.114176E+04
  0.106751E+04  0.102991E+04  0.110927E+04  0.101118E+04  0.104475E+04
  0.104928E+04  0.103831E+04  0.930903E+03  0.968112E+03  0.934791E+03
  0.979705E+03  0.952782E+03  0.940582E+03  0.956763E+03  0.873679E+03
  0.873245E+03  0.932861E+03  0.906989E+03  0.888141E+03  0.865821E+03
  0.839173E+03  0.855407E+03  0.839656E+03  0.850966E+03  0.797723E+03
  0.784301E+03  0.817942E+03  0.806588E+03  0.772983E+03  0.797675E+03
  0.745588E+03  0.784774E+03  0.776142E+03  0.756732E+03  0.771707E+03
  0.388920E+01  0.104280E+02  0.157278E+02  0.187425E+02  0.196304E+02
  0.189484E+02  0.172882E+02  0.151361E+02  0.128411E+02  0.106267E+02
  0.861921E+01  0.687585E+01  0.540920E+01  0.420518E+01  0.323589E+01
  0.246793E+01  0.186755E+01  0.140347E+01  0.104821E+01  0.778542E+00
  0.575364E+00  0.423284E+00  0.310116E+00  0.226346E+00  0.164631E+00
  0.119361E+00  0.862829E-01  0.622007E-01  0.447258E-01  0.320838E-01
  0.229641E-01  0.164024E-01  0.116928E-01  0.832014E-02  0.591005E-02
  0.419124E-02  0.296772E-02  0.209831E-02  0.148154E-02  0.104469E-02
  0.735726E-03  0.517522E-03  0.363621E-03  0.255212E-03  0.178938E-03
  0.125336E-03  0.877076E-04  0.613206E-04  0.428349E-04  0.298970E-04
  0.208502E-04  0.145298E-04  0.101178E-04  0.704052E-05  0.489581E-05
  0.340218E-05  0.236273E-05  0.163984E-05  0.113745E-05  0.788520E-06
  0.546326E-06  0.378319E-06  0.261841E-06  0.181134E-06  0.125241E-06
  0.865548E-07  0.597909E-07  0.412844E-07  0.284937E-07  0.196575E-07
  0.135560E-07  0.934462E-08  0.643909E-08  0.443532E-08  0.305398E-08
  0.210210E-08  0.144640E-08  0.994899E-09  0.684111E-09  0.470256E-09
  0.323152E-09  0.221997E-09  0.152460E-09  0.104675E-09  0.718461E-10
  0.492997E-10  0.338196E-10  0.231941E-10  0.159028E-10  0.109009E-10
  0.747036E-11  0.511819E-11  0.350581E-11  0.240083E-11  0.164375E-11
  0.112515E-11  0.770008E-12  0.526849E-12  0.360402E-12  0.246490E-12
  0.168548E-12  0.115229E-12  0.787625E-13  0.538262E-13  0.367780E-13
  0.251249E-13  0.171610E-13  0.117194E-13  0.800187E-14  0.546268E-14
  0.372863E-14  0.254461E-14  0.173630E-14  0.118457E-14  0.808030E-15
  0.551099E-15  0.375810E-15  0.256237E-15  0.174684E-15  0.119071E-15
  0.811513E-16  0.553002E-16  0.376790E-16  0.256693E-16  0.174853E-16
  0.119090E-16  0.811009E-17  0.552230E-17  0.375978E-17  0.255948E-17
  0.174217E-17  0.118571E-17  0.806891E-18  0.549039E-18  0.373545E-18
  0.254118E-18  0.172855E-18  0.117566E-18  0.799529E-19  0.543679E-19
  0.369664E-19  0.251321E-19  0.170846E-19  0.116129E-19  0.789285E-20
  0.536396E-20  0.364500E-20  0.247667E-20  0.168267E-20  0.114312E-20
  0.776506E-21  0.527425E-21  0.358211E-21  0.243265E-21  0.165190E-21
  0.112163E-21  0.761524E-22  0.516988E-22  0.350948E-22  0.238216E-22
```

```
 TEST DATA SET 2      ***  SUM OF EXPONENTIALS  ***
 ND                        3.
 NNL                       2.
 NL         1              2.
 NL         2              3.
 NGAM                      1.
 IWT                       1.
 IFORMY
 (4E20.6)
 SAMET                     1.
 DOUSOU     2              1.
 DOADEX
TTTTTTTTTTTTTTTTTTTT
 IPRITR     2              3.
 IPLFIT     2              1.
 MTRY
  5   5   5   5   5   5   5   5   5   5   5   5   5   5   5   5   5   5   5   5
 MXITER     2             20.
 IUSER      1              1.
 PLMNMX     1             0.0
 PLMNMX     2         0.1E+03
 PNMNMX     1             0.2
 DOUSIN                   -1.
 LAST                      1.
 END
 NINTT     1
 NSTEND   50               0.           98.
        0.392630E+02        0.378621E+02        0.341216E+02        0.330735E+02
        0.315259E+02        0.299061E+02        0.291313E+02        0.275977E+02
        0.282081E+02        0.265212E+02        0.257173E+02        0.253186E+02
        0.249112E+02        0.249183E+02        0.234907E+02        0.234972E+02
        0.235639E+02        0.232833E+02        0.228483E+02        0.223709E+02
        0.220083E+02        0.220761E+02        0.217859E+02        0.217283E+02
        0.214995E+02        0.222927E+02        0.207859E+02        0.212752E+02
        0.207515E+02        0.207152E+02        0.206974E+02        0.208856E+02
        0.200046E+02        0.206331E+02        0.209575E+02        0.205997E+02
        0.210100E+02        0.204239E+02        0.198299E+02        0.204870E+02
        0.201594E+02        0.206341E+02        0.203827E+02        0.204851E+02
        0.204052E+02        0.205013E+02        0.198743E+02        0.209660E+02
        0.201976E+02        0.199987E+02
        0.494667E+02        0.445952E+02        0.440464E+02        0.411485E+02
        0.391268E+02        0.369257E+02        0.356110E+02        0.334534E+02
        0.327403E+02        0.316328E+02        0.303412E+02        0.293192E+02
        0.282534E+02        0.279143E+02        0.274521E+02        0.262449E+02
        0.262892E+02        0.257003E+02        0.250714E+02        0.245336E+02
        0.248013E+02        0.235956E+02        0.239378E+02        0.235334E+02
        0.242268E+02        0.211070E+02        0.216601E+02        0.223464E+02
        0.226617E+02        0.224353E+02        0.220892E+02        0.215333E+02
        0.210741E+02        0.209988E+02        0.212725E+02        0.217476E+02
        0.215578E+02        0.203018E+02        0.209608E+02        0.206410E+02
        0.209450E+02        0.198926E+02        0.208796E+02        0.199937E+02
        0.211655E+02        0.203414E+02        0.196332E+02        0.204600E+02
        0.201782E+02        0.199145E+02
        0.299651E+02        0.259961E+02        0.216722E+02        0.192823E+02
        0.165780E+02        0.142475E+02        0.119444E+02        0.104221E+02
        0.863121E+01        0.838118E+01        0.730875E+01        0.687681E+01
        0.580214E+01        0.501164E+01        0.484633E+01        0.346130E+01
        0.375805E+01        0.354401E+01        0.323136E+01        0.250307E+01
        0.259458E+01        0.261215E+01        0.197752E+01        0.198894E+01
        0.154513E+01        0.176231E+01        0.107486E+01        0.118563E+01
        0.890578E+00        0.129350E+01        0.744488E+00        0.113555E+01
        0.700696E+00        0.965748E+00        0.931408E+00        0.644300E+00
        0.894842E+00        0.477920E+00       -0.596124E-01        0.225075E+00
        0.720417E+00        0.648557E+00        0.627091E+00        0.399277E+00
       -0.106968E+00        0.708857E+00        0.611183E+00        0.945263E+00
        0.165471E+00        0.343607E+00
```

3.3.2 Composition of a Data Set

Below is a listing of all the types of cards that make up a Data Set, listed in order of input.

Card	FORMAT	Input variables	Necessary when
1	80A1	ITITLE(K),K=1,80	always
set 2	1X,6A1,I5,E15.6	Control Variables	optional — see Sec 3.4
3	1X,6A1,I5,E15.6	"END" card	always
4a	1X,6A1,I5	"NINTT",NINTT	regular intervals for t_k
set 4b	1X,6A1,I5,2E15.6	"NSTEND", NT, TSTART, TEND NINTT times	regular intervals for t_k
5a	1X,6A1,I5	"NY", NY	nonreg. interv. for t_k
set 5b	IFORMT	T(K,1),K=1,NY	" " " "
set 6	IFORMY	Y(K,1),K=1,NY	SIMULA=.FALSE.
set 7	IFORMW	SQRTW(K,1), K=1,NY	IWT=4
sets 4–7	(see section 4.1.2)		ND > 1
sets 8–11b	(see section 4.2)	input from USER subprograms	

Card 1 contains a heading of up to 80 characters, which will be printed at key places in the output. Card 1 is always necessary, but it can be blank. (See Lines 1 and 75 of File 3).

Card Set 2 contains optional Control Variables. They are described in Secs 3.4 and 4.1. (See Lines 2–7 and 76–98 of File 3.)

Card 3 contains the characters END in columns 2–4. (See Lines 8 and 99 of File 3.) It is usually alright to leave the rest of the card blank. However, the actual FORMAT is

```
FORMAT (1X,6A1,I5,E15.6)
```

and two other (unused) variables are input from the I5 and E15.6 fields. Therefore in the rare case that your computer does not allow blank fields, it may be necessary to put legal constants (e.g., 0 and 0.) in these two fields. Card 3 marks the end of Card Set 2 and is always needed, even if there is no Card Set 2.

Card 4a and Card Set 4b are a convenient alternative for inputting the t_k in Eq (3.1–2) when the t_k can be input in NINTT groups of equally spaced t_k. (See Lines 9–10 and 100–101 of File 3.) Card 4a contains the characters NINTT in columns 2–6 and the value of NINTT right justified in columns 8–12 (FORMAT I5). Card Set 4b has NINTT cards, one for each group, all in the same

```
FORMAT (1X,6A1,I5,2E15.6)
```

with the characters NSTEND in columns 2–7, followed by NT, TSTART, and TEND, where

NT = the number of t_k in the group (NT ≥ 2),
TSTART = the first t_k in the group,
TEND = that last t_k in the group.

For example, if the t_k were

1,2, 4,6,8,10, 20,30,40,50, 100,200,300,400,500

then you could input NINTT=4 and

NT	TSTART	TEND
2	1.	2.
4	4.	10.
4	20.	50.
5	100.	500.

Card 5a and Card Set 5b are used when the t_k are to be read in. Card 5a contains the characters NY in columns 2 and 3 and the value of N_y in Eq (3.1–1) right-justified in columns 8–12 (FORMAT I5). Card Set 5b contains the N_y values of t_k in Eq (3.2–2). The FORMAT is specified by the Control Variable IFORMT (Sec 3.4.1).

Card Set 6 contains the N_y values of the y_k in Eqs (3.1–1) and (3.1–2). (See Lines 11–42 and 102–140 of File 3.) The FORMAT is specified by IFORMY. Card Set 6 is necessary unless you are simulating your data with SIMULA=T.

Card Set 7 contains the N_y values of the least-squares weights, w_k in Eq (3.2.1–1). The FORMAT is specified by IFORMW. Card Set 7 is needed when IWT=4. See Sec 4.1.3.4 for other ways of specifying the w_k with IWT.

Card Sets 8 – 11b are for possible input from `USER` subprograms and are usually not needed. (See Lines 43–74 of File 3.) They are described in Sec 4.2. However, you only have to read Sec 4.2 if you are using a Control Variable that leads to input in a `USER` subprogram (and the description of the Control Variables in Sec 4.1 will tell you this) or if you are making your own modifications of `USER` subprograms that result in input.

3.4 Essential Control Variables

The 39 Control Variables make SPLMOD very flexible. They are all set to default values in the `BLOCK DATA` subprogram. Therefore, it is only necessary to input a Control Variable in Card Set 2 if you wish to change its value. Card Set 2 can be empty.

You can reset the default values in the `BLOCK DATA` subprogram to values that you will most often use. This can save you from inputting them in Card Set 2 every time. You should keep a listing of the current version of your `BLOCK DATA` subprogram so that you know what the default values are. However, SPLMOD also prints out the final values of the Control Variables that will be used for each data set; so this gives you a final check on their values (Sec 3.6.2).

A Control Variable is input with one or two cards. The `FORMAT` for the first card is always the same, `(1X,6A1,I5,E15.6)`. The name of the Control Variable must be in the `6A1` field, left justified (i.e., starting in column 2) followed by enough blanks to fill the `6A1` field if the name is less than six characters. An error message is printed if SPLMOD is expecting the first card for a Control Variable and the name of a Control Variable is not properly located in columns 2–7, and the run will be eventually aborted.

Most Control Variables must be input with one card. These Control Variables are either undimensioned or one-dimensional arrays. The `I5` field contains the single subscript of the Control Variable, if it has one. The `E15.6` field specifies the value. If the Control Variable is of type `INTEGER`, then the value (converted to `REAL`) must be in the `E15.6` field. For `LOGICAL` Control Variables , only the values `1.0` (for `.TRUE.`) and `-1.0` (for `.FALSE.`) are allowed.

Eight Control Variables require two cards. In this case the first card still has the same `FORMAT`, but the `I5` and `E15.6` fields are not used (although they are read). In the descriptions of the Control Variables below and in Sec 4.1, the name of the Control Variable is followed by its `DIMENSION` specification (if it is an array) followed by the `FORMAT` and Fortran input list specification for the

second card (if the Control Variable requires two cards) and finally instructions for its use. All Control Variables are also listed in Sec 8.2 together with the type (`REAL`, `INTEGER`, or `LOGICAL`), `DIMENSION` (if any), `FORMAT` (where necessary), default values, and the number of the section where the Control Variable is defined.

I have adhered throughout SPLMOD to the standard naming convention, whereby all `INTEGER` variables begin with letters in the range `I-N` and all `REAL` variables with letters in the ranges `A-H` and `O-Z`. Therefore the descriptions only indicate when the Control Variables are `LOGICAL`; everything else is clear, since there are no `DOUBLE PRECISION` Control Variables. For `LOGICAL` variables, I use the standard abbreviations (that are also used on input and output) of `T` for `.TRUE.` and `F` for `.FALSE.`.

Only the essential Control Variables are listed in this section. More Control Variables are described in Sec 4.1.

3.4.1 Input control

`LAST = T`	if this is the last data set in the input data deck.
`= F`	if one or more data sets follow this one.

A Data Set always has Card 1 as its first card and contains all the cards and Card Sets from the range Card 1–Card Set 11 that are necessary for an analysis of a set of y_k. An input data deck may consist of one or more Data Sets. After SPLMOD has completed the analysis of a Data Set, it stops if `LAST=.TRUE.` in the Data Set just analyzed; if `LAST=.FALSE.`, it reads in the next Data Set and analyzes it.

The settings of all Control Variables are preserved from the analysis of one Data Set to the next. For example, if the default value of `LAST` were set to `.TRUE.` in `BLOCK DATA` and you wanted to analyze five Data Sets , you would input `LAST` in Data Set 1 (changing it to `.FALSE.`), not input it in Data Set 2, 3, and 4 (leaving it `.FALSE.`), and input it in the last Data Set (changing it to `.TRUE.`). This is illustrated for two Data Sets in Lines 7 and 98 of File 3.

In this way, a lot of input in Card Set 2 is often avoided in Data Sets after the first one. Note that this preservation of values only applies to the Control Variables. The necessary data on Card 1 and Card 3 - Card Set 11b must be in every Data Set.

`IFORMY` (70)	`FORMAT(1X,70A1) (IFORMY(J),J=1,70)` The second card must contain a Hollerith string of characters making up a legal Fortran `FORMAT` specification (enclosed in parentheses) for the y_k in Card Set 6. It is illustrated in Lines 82–83 in File 3.
`IFORMT` (70)	`FORMAT(1X,70A1) (IFORMT(J),J=1,70)` This is defined in the same way as `IFORMY` above, except that it specifies the `FORMAT` for the input of the t_k in Card Set 5b.
`IFORMW` (70)	`FORMAT(1X,70A1) (IFORMW(J),J=1,70)` This is defined in the same way as `IFORMY` above, except that it specifies the `FORMAT` for the input of the w_k in Card Set 7.

3.4.2 Setting N_λ and NNL, the maximum number of N_λ

`NNL`	The maximum number of not necessarily different N_λ values, that will be searched for. The following restriction must be satisfied: $1 \leq$ `NNL` ≤ 10.

`NL` (10)	`NL(J),J=1,...,NNL`, the not necessarily different N_λ values, that will be used in `NNL` least squares analyses. If no `NL(J)` is input in Card Set 2, SPLMOD will set `NL(J)=J` for `J=1,...,NNL`.

To preserve objectivity and check for systematic experimental errors or unexpected components, it is always a good idea to choose the maximum N_λ 1 larger than the one you expect. There is in principle no restriction for N_λ, except that N_λ has to be greater than 0, but since most of the computer time is usually spent looking for components that are not there, it is a great waste to put in a value of N_λ that is too large for the data to determine (e.g., $N_\lambda = 4$ for 40 points with equally spaced t_k and 5% experimental error). On the other hand, if one knows a priori that the only reasonable solution is for certain N_λ, e.g., $N_\lambda = 3$, then it is possible to input `NNL=1` and `NL(1)=3`.

3.4.3 Specifying boundary conditions for parameters

`PLMNMX` (2)	Constraints for the linear parameters α_j and γ_μ .

The following restrictions are held during all the least squares analyses:

$$\texttt{PLMNMX}\,(1)\cdot|\texttt{YMAX}| \leq \alpha_j, \gamma_\mu \leq \texttt{PLMNMX}\,(2)\cdot|\texttt{YMAX}|\,, \qquad \begin{array}{l} j = 1,\ldots,N_\lambda, \\ \mu = 1,\ldots,N_\gamma \end{array}$$

where $|$`YMAX`$|$ is the absolute maximum value of the data y_k.

This allows you, for example, to constrain the amplitudes α_j to be nonnegative by setting `PLMNMX(1)=0.`, if there is a priori knowledge that $\alpha_j > 0$ for $j > 0$; e.g., if the α_j correspond to concentrations or populations. Care has to be taken in choosing `PLMNMX` for some special functions of $f(\lambda, t)$, e.g., when $f(\lambda, t)$ is a convolution the size of the α_j also depends on the spacing of the t_k and the size of the response function. The version of `USERIN` shows how `PLMNMX` can be recalculated in such a case (See Sec 4.1.4).

`PNMNMX` (2)	Constraints for the nonlinear parameters λ_j.

During all the least squares analyses the λ_j are constrained to be within these limits:

$$\texttt{PNMNMX}\,(1)/(t_{N_y} + d) \leq \lambda_j \leq \texttt{PNMNMX}\,(2)/(t_1 + d), \qquad j = 1,\ldots,N_\lambda$$

where $d = (t_5 - t_1)/4 - t_1$ with reordered t_k if necessary in monotonically increasing order with k.

This prevents the λ_j from entering physically unfeasible (or, at least for the range of the t_k physically indeterminable) regions, and from perhaps causing arithmetic overflows. It also gives the solution a chance to rebound back into a feasible region after a bad iteration step.

The default values are `PNMNMX(1)=.02` and `PNMNMX(2)=2.08`, however if `NGAM` > 0 `PNMNMX(1)` should be input with a factor of ten bigger than the default value. In very rare cases it may occur that these limits are too strict, in particular when there is a component with a very large amplitude and $\lambda_j >$ `PNMNMX(2)`. Therefore you can input `PNMNMX(1)` with a smaller value (up to a factor of 10) or `PNMNMX(2)` larger (up to a factor of 3).

3.4.4 Lines per page of output

`LINEPG`	The number of lines per page available on your output device.

It is used to plan certain plotted output (of the residuals) so that plots are not broken up going from one page to the next.

3.5 Changing DIMENSION Specifications

If your problem is large, it may be necessary to increase the **DIMENSION** specifications of some arrays before you can use SPLMOD. Throughout SPLMOD there are checks to make sure that these maximum allowable dimensions are not exceeded; so one simple way to see if your problem is too large is to try running SPLMOD and see if it stops with an error message telling you this. If SPLMOD doesn't stop with an error message, then you don't have to read Sec 3.5. However, it is probably worthwhile to first check to make sure that at least the obvious requirements below (e.g., **NYMAX**) are satisfied. Furthermore, if high-speed storage is expensive, then it would be worthwhile to check below to see if your problem will always be small enough to permit you to reduce some of the **DIMENSION** specifications.

The **DIMENSION** specifications are in terms of the eight **DIMENSION** Parameters **NONLMX**, **NGAMMX**, **NLINMX**, **NDMAX**, **NLNDMX**, **NYMAX**, **NZMAX**, and **MTRYMX**. They must satisfy the following requirements:

$$\begin{array}{lcll}
\mathtt{NONLMX} & \geq & \max\{\mathtt{NL(J)}+1\}, \quad \mathtt{J}=1,\ldots,\mathtt{NNL} & (1)\\
\mathtt{NGAMMX} & \geq & N_\gamma & (2)\\
\mathtt{NLINMX} & \geq & \mathtt{NONLMX}+N_\gamma & (3)\\
\mathtt{NDMAX} & \geq & N_D & (4)\\
\mathtt{NLNDMX} & \geq & \mathtt{NLINMX}*\mathtt{NDMAX}+\mathtt{NONLMX} & (5)\\
\mathtt{NYMAX} & \geq & \max\{\mathtt{NY(J)},\mathtt{NLNDMX}\}, \quad \mathtt{J}=1,\ldots,N_D & (6)\\
\mathtt{NZMAX} & \geq & \mathtt{NB}=\mathtt{NZ}+2\geq 20 & (7)\\
\mathtt{MTRYMX} & \geq & \max\{\mathtt{MTRY(I,J)}\}, \ \mathtt{I}=1,\ldots,\mathtt{NNL}, \ \mathtt{J}=1,2 & (8)
\end{array}$$

By $\max\{J,K\}$ is meant the maximum of J and K. All other quantities on the right-hand sides can be found in the index in Sec 8.

SPLMOD is designed so that you can change the **DIMENSION** specifications in two easy steps in the **MAIN** subprogram: First change the values of the eight **DIMENSION** parameters in the **DATA** statement in Lines **MAIN0140**–**MAIN0147**. Then change the **DIMENSION** statements in Lines **MAIN0064**–**MAIN0081** according to the specifications shown in lines **MAIN0085**–**MAIN0113**. No other changes in the **MAIN** subprogram or any of the other subprograms are needed. However if you will use, e.g., recursion formulas, to evaluate $f(\lambda,t)$, as shown in subprogram **USERFN**, and you will therefore need additional **COMMON** or **DIMENSION** specifications, then you will have to change these too. You will have to take care of extra **COMMON** or correct **DIMENSION** in the user supplied subprograms anyway.

3.6 Interpreting the output

3.6.1 Output from run with the test data

This section lists the output from a run on the *VAX 8650* at EMBL with the test data in File 3 (Sec 3.3.1). Before you compare these too closely with your output, read Sec 3.6. It indicates the many parts that are strongly machine-dependent. Section 3.6 also uses examples from this output to illustrate important general guidelines and points to check in the entire output. For the purposes of Sec 3.6, it is not necessary to know the meanings of the Control Variables in the test data; see Sec 4.1 for more details. Section 4.1.5 describes useful Control Variables that can increase or decrease the amount and detail of the output. The data for *Test Data Set* 1 are simulated fluorescence decay data, where a two-component exponential decay is convoluted with an exciting lamp intensity and a constant background is added. The parameters used were $\lambda_1 = .02$, $\lambda_2 = .08$, and $\alpha_1/\alpha_2 = 1$; the α_j and γ_1 were adjusted so that $y(t) = 10^4$ at its maximum. The added noise was normal distributed with zero mean and variance $y(t)$. Test Data Set 2 contains 3 different data sets, which are analyzed simultaneously, each involving a sum of two exponentials with different amplitudes α_j and background γ_1 but the same time constants λ_j.

3.6.2 Output of the input data

The first line of *Page 1* gives the version and date of SPLMOD and the heading from Card 1 (see Sec 3.3.2). Then Card Set 2, Card 3, Card 4a, and Card Set 4 (for Test Data Set 1, which is analyzed first) are printed out as they are read in.

Page 2 contains the output of the response function, which was input in Card Set 8 (for Test Data Set 1 only) by **USER** subprogram **USERIN**.

Page 3 contains the final values of the Control Variables, after the changes specified in Card Set 2 have been made. These are the values that will be used throughout the analysis of Test Data Set 1. It is always a good idea to check and make sure that these values are correct. They are output in six groups (in alphabetical order): undimensioned and dimensioned **REAL**, **INTEGER**, and **LOGICAL**, respectively.

Finally, the data set number and the values of t_k and y_k that will be used in Eqs (3.1–1) and (3.2–1), are printed on *Page 4*.

3.6.3 Machine-dependent parameters — PRECIS

On *Page 4* you will find three machine-dependent parameters, `SRANGE` and `RANGE`, which you previously set in `BLOCK DATA` (see Sec 2.3.1), and `PRECIS`, which is computed in SPLMOD as approximately 10 times the relative machine precision; i.e., `PRECIS` is approximately the smallest positive number so that `1.0+0.1·PRECIS` is computed as greater than `1.0` (in `DOUBLE PRECISION` for Version `3DP` and `REAL` for Version `3SP`). This is the only line that explicitly tells you whether you are using Version `3SP` or `3DP`; if you were using Version `3SP`, then `PRECIS` and `RANGE` would have `E` instead of `D` exponents, e.g., `1.86E-16` and `1.00E+35`. Thus `PRECIS=1.86D-16` means that there are approximately 16 significant figures in the `DOUBLE PRECISION` arithmetic. If `PRECIS` is greater than `1.00E-12`, then you should switch from Version `3SP` to Version `3DP`; you should generally have at least 12-significant-figure arithmetic (see also Sec 2.2).

3.6.4 Solutions

SPLMOD performs `NNL` analyses assuming N_λ=`NL(J),J=1,...,NNL` components in Eq (3.1–1). If `IWT`=2,3, or 5, SPLMOD first performs a preliminary analysis, as explained in Sec 4.1.3.4, with all the $w_k = 1$, to get a smooth fit through the y_k (these `NNL` analyses are printed on *Pages 5–9*); this smooth fit is then used to compute less biased w_k (their square-roots are printed on *Page 10*) and then the final weighted analyses are made (they are printed on *Pages 11–21*).

3.6.4.1 Heading

The first line of output for an analysis with a new N_λ contains the heading from Card 1 and one of the following phrases:

`PRELIMINARY UNWEIGHTED` *nlam* `COMPONENTS ANALYSIS`

or

`FINAL` *nlam* `COMPONENTS ANALYSIS`

where *nlam* represents the actual value of N_λ.

The first phrase only occurs when `IWT`=2,3, or 5. For each analysis with a certain N_λ a best solution is also chosen, which has an additional phrase printed

`BEST SOLUTION`

at the right upper corner of the page, just before one of the lines mentioned above.

3.6.4.2 Iterations

Following the heading, there will be the output of the results of each iteration of the least squares fit to the data, as specified through the Control Variable `IPRITR`. There are options to print out all iteration steps, the first and the last one, the last one only, and no iterations at all (See Sec 4.1.5.2 for a detailed description of `IPRITR`). In any case, if errors occur during the iterations (see Sec 5.3 for the description of errors), they will be printed out, e.g., on *Pages 5* and *8*. So you can keep track of what was going wrong while iterating, even if you don't print out any iteration steps. If iterations are printed out, as on *Pages 11, 14, 15,* and *18, 19*, they will be all headed with the same terms defined below:

`ITR`	the number of the iteration
`VARIANCE`	`VAR` in Eq (3.2–1). An asterisk next to the value means that the variance has increased (e.g., see *Page 14*). This can occur if the normal equations matrix is nearly singular, or sometimes at the end of the analysis when the variance is already very close to the minimum.
`DAMPING Q`	the usual damping factor used to modify the length of a *Gauß* step in nonlinear least squares (e.g., q on p. 155 of [17]). A value of `Q` consistently near 1 indicates the problem is nearly linear, and a `Q` < 0.01 indicates a highly nonlinear situation. The value of `Q` giving the minimum `VAR` is first estimated by fitting a quadratic polynomial usually to $\mathtt{VAR}_{(\mathtt{Q}=0)}$, $\partial\mathtt{VAR}/\partial\mathtt{Q}_{(\mathtt{Q}=0)}$, $\mathtt{VAR}_{(\mathtt{Q}=1)}$. If `VAR` for `Q` turns out to be larger than any of the `VAR`'s used for the quadratic fit, the `Q` with the lowest `VAR` is used and an asterisk is printed next to it (as on *Page 15*).
`GAMMA MU`	current value of γ_μ in Eq (3.1–1).
`ALPHA J, LAMBDA J`	current values of α_j, λ_j in Eq (3.1–1).

If one of the above mentioned parameters γ_μ, α_j, or λ_j was temporarily held out of the regression, either because it would violate a constraint or lead to a nearly singular normal equations matrix, an asterisk is printed next to it (as on *Page 26*).

These iterations will be repeated until convergence is reached (see Sec 4.1.7.3 for `MCONV`, `NABORT`, and the description of convergence criteria), or a maximum number of iterations is performed (see Sec 4.1.7.3 for the description of `MXITER`).

There is also the possibility of doing a very thorough and reliable grid search in parameter space for least squares solutions to Eq (3.2–1), for a single N_λ. As specified through Control Variable `MTRY` (see Sec 4.1.7.4 for a detailed description of `MTRY`), SPLMOD will do a systematic search with different starting values for the λ_j, whereby SPLMOD keeps track of the path through parameter space during the iterations, to avoid starting values which were already tried during other iterations.

3.6.4.3 Table with solution for a single N_λ

It is the most important output and is always printed. There is also an option to print this table after each try with new starting values, or only when the variance `VAR` of a new solution is lower than a previous one (see Sec 4.1.5.2 for a complete description of Control Variable `IPRINT`). The default output of the table is for the best solution only, for a single N_λ, as on *Pages 6, 7,* and *9*, where the criterion *best* means the solution having the lowest variance `VAR` found in previous `MTRY` analyses.

The solution is printed out in 3 parts as described below:

1) the standard deviation of the fit:

`STANDARD DEVIATION OF FIT = STDFIT =` $[\texttt{VAR}/(N_y - 2N_\lambda - N_\gamma)]^{1/2}$,
for `VAR` see Eq (3.2–1).

Unless a singular or nearly singular normal equations matrix occurs, the following 2 parts are printed too:

2) the parameters

`GAMMA, ALPHA, LAMBDA` — the γ_μ, α_j, and λ_j of the solution.

`STD. ERROR` — $\sigma(\gamma_\mu)$, $\sigma(\alpha_j)$, and $\sigma(\lambda_j)$, i.e., the standard error estimates of the respective γ_μ, α_j, and λ_j.

These are obtained in the usual way [17] from the estimated variance-covariance matrix. Thus, ignoring the nonlinearity, the 95% confidence intervals are approximately $\pm 2\sigma(\gamma_\mu)$, $\pm 2\sigma(\alpha_j)$, and $\pm 2\sigma(\lambda_j)$ respectively.

If you are interested in a more detailed error analysis, you can always use the γ_μ, α_j, and λ_j and the data as input for a straightforward, but time consuming, exploration of the contour lines of the variance surface to get even more accurate confidence regions. (See also Control Variable `DOUSOU`.)

`PERCENT` — $100\,\sigma(\gamma_\mu)/|\gamma_\mu|$ and similarly for α_j and λ_j.

3) the correlation matrix

`CORRELATION COEFFICIENTS` — a table of the usual estimates [17] of the coefficients of correlation between parameter pairs.

3.6.4.4 Plot of the weighted residuals

The *residuals* are the N_y values of the left-hand sides of Eq (3.1–1) minus the right-hand sides. The N_y weighted residuals are simply the residuals multiplied by ${w_k}^{1/2}$; so the sum of their squares is just `VAR` in Eq (3.2–1). When the residuals are to be output, you have to specify via Control Variable `IPLRES(2)` (See Sec 4.1.5.2 for a detailed description of the possibilities of `IPLRES`).

The weighted residuals are plotted, e.g., on *Page 13* for the best solution found in a 1 component analysis. The maximum (`8.5E+00`) and minimum (`-1.3E+01`) values of the weighted residuals are printed on the first line of *Page 13*, and the top and bottom dashed lines of the plot correspond to these values. The dashed line in between corresponds to zero. The abscissa is simply the subscript of the residuals, k.

The term `RANDOM RUNS PROB.` is also printed on the first line. This is the approximate probability that a series of N_y zero-mean uncorrelated random numbers with the same number of positive and negative values as the residuals would have no more runs than the residuals [12, p. 95]. The number of runs is one plus the number of times the residuals change sign as k in Eq (3.2–1) goes from 1 to N_y. Therefore `RANDOM RUNS PROB.` near zero (say, less than about 0.05) means a very nonrandom distribution of residuals with an unusually small number of runs, and this can indicate an inadequate fit to the data, e.g., due to systematic experimental errors. The significance of `RANDOM RUNS PROB.` is quite weak, however, unless N_y is at $O(100)$. In fact, unless there are at least 10 positive and 10 negative residuals, the approximation [12, p. 97] itself gets so poor that a `-1.0` is printed in place of `RANDOM RUNS PROB.` to indicate this. (Otherwise, `RANDOM RUNS PROB.` is always between `0.0` and `1.0`.) If the plots of the residuals and the fit to the data (see Sec 3.6.4.5) look reasonable and N_y is less than $O(100)$, it may be just a chance occurrence of a small number of runs.

Also note that `RANDOM RUNS PROB.` is only meaningful if the ordering of the residuals is meaningful, e.g., if t_k in Eq (3.1–1) is varying monotonically with k.

The plot of the residuals can be very useful for several reasons:

- You can easily spot outliers, which may have been caused by gross errors in inputting your data.
- Together with `RANDOM RUNS PROB.` and the plot of the fit (Sec 3.6.4.5), you can recognize a systematic lack of fit of your model to the data.

What `RANDOM RUNS PROB.` cannot detect is a systematic trend in the magnitudes of the residuals. This can happen, e.g., if the wrong type of weighting is used.

3.6.4.5 Plot of the fit to the data

SPLMOD also has an option (see Control Variable `IPLFIT`) to plot the data, y_k (printing character `O`), and the fit to the data (`X`), i.e., the right-hand side of Eq (3.1–1), as shown on *Pages 32–34* in the analysis of Test Data Set 2. The fit values $\hat{y}_k$ are printed under `ORDINATE`, and the t_k are printed under `ABSCISSA`, see Eq (3.1–1). Asterisks (`*`) are plotted when the `X` and `O` characters coincide.

This plot helps you judge the fit from a more natural perspective, but is not as sensitive as the plot of the residuals at seeing trends or outliers. Each line in the plot has a resolution of 108 spaces. Therefore, each space separating the `X` and `O` on a line corresponds to a difference of about 1% of the total range of the y_k.

Sec 4.1.5.2 tells you more about the Control Variable `IPLFIT`. It also tells you how to suppress these plots completely or how to plot interesting parts only in case of a very large N_y causing too much printout (but the residuals are plotted compactly, and you should always plot them).

3.6.5 Solution with `BEST FIT TO DATA`

After a complete analysis we are faced with the problem of deciding which of the solutions is the *best*. The following procedure was found to be very effective [8]. Consider two solutions with $N'_\lambda < N_\lambda$ and `VAR` and `VAR'`, where `VAR` is defined

in Eq (3.2–1). In the unusual case that `VAR` > `VAR'`, the N_λ solution is rejected. Otherwise the program computes:

$$\begin{aligned} \mathtt{PNG}\,(N'_\lambda|N_\lambda) &\equiv 1-\mathtt{PNG}\,(N_\lambda|N'_\lambda) \qquad (1)\\ &= P\,[\,F\,|\,2(N_\lambda-N'_\lambda)\,,N_y-2N_\lambda-N_\gamma\,] \end{aligned}$$

where

$$F = \frac{N_y-2N_\lambda-N_\gamma}{2(N_\lambda-N'_\lambda)}\cdot\frac{(\mathtt{VAR}'-\mathtt{VAR})}{\mathtt{VAR}}$$

and $P\,[\,F|\nu_1,\nu_2]$ is Fisher's F-distribution [10] with ν_1 and ν_2 degrees of freedom. `PNG(K|J)` is the approximate probability that the solution with N_λ = `K` is worse (or less likely) than the solution with N_λ = `J`.

These terms are printed on *Pages 10* and *22*, headed by:

```
FOUND BEST FIT TO DATA WITH nlam COMPONENTS.
PROBABILITIES THAT OTHER SOLUTIONS ARE WORSE:
```

and the above described `PNG(K|J)` are following. The *best* solution is chosen as the N_λ solution such that `PNG`$(N'_\lambda|N_\lambda) > 0.5$ for all other N_λ. You should not blindly accept the best solution on the last page and look at nothing else. Of course one should impose the usual significance levels and consider any alternate N'_λ solution with `PNG`$(N'_\lambda|N_\lambda) \leq 0.95$ to be still a significant probability as, e.g., on *Page 10* the `PNG` for the $N_\lambda = 2$ solution.

3.6.6 Final weighted analysis

The parameters γ_μ, α_j, and λ_j for the `BEST FIT TO DATA` found in all `NNL` different `PRELIMINARY UNWEIGHTED ANALYSES` which are being used to compute the weights, are printed on *Page 10*, as described in Sec 3.6.5, headed by the line: `PARAMETERS USED TO GENERATE WEIGHTS`. Also printed on *Page 10* are `ERRFIT`, an optional safety margin that is used to compute the weights (it is discussed in Sec 4.1.3.4), and the `SQUARE ROOTS OF THE LEAST SQUARES WEIGHTS`.

In fact, the sole purpose of the `PRELIMINARY UNWEIGHTED ANALYSIS` is to determine these weights, being used for the weighted analysis on *Pages 11–22*. If you specified the weights in the input data, i.e., if `IWT=1` or `4` (see Secs 4.1.3.4 and 4.1.4), then the entire preliminary analysis on *Pages 4–9* would be missing, as with Test Data Set 2 on *Pages 23–45*.

It is always a good idea to compare the solutions for the unweighted and weighted analyses. It is reassuring when they do not differ drastically from one another, since this indicates that they are at least relatively stable to changes in weighting.

The final weighted analysis is on *Pages 11–22*. (You should not expect exact agreement with your output.) The format is the same as that discussed for *Pages 5–9*, except that **PRELIMINARY UNWEIGHTED ANALYSIS** in the heading is replaced by **FINAL ANALYSIS**, and the solution with the **BEST FIT TO DATA** is again printed at the end of the analysis, on *Page 22*. *Pages 23–45* contain the analysis of Test Data Set 2. They are mainly used in Sec 4 to illustrate more possibilities of the Control Variables.

```
SPLMOD - VERSION 3DP (JUN 1988)          TEST DATA SET 1      ***  CONVOLUTION  ***

REFERENCES - S.W.PROVENCHER AND R.H.VOGEL (1983) IN PROGRESS IN SCIENTIFIC COMPUTING, VOL. 2, PAGES 304-319,
                                                 P.DEUFLHARD AND E.HAIRER EDS., (BIRKHAEUSER, BOSTON)
           - R.H.VOGEL                      (1988) SPLMOD USERS MANUAL, EMBL TECHNICAL REPORT DA09,
                                                 (EUROPEAN MOLECULAR BIOLOGY LABORATORY, HEIDELBERG, F.R. OF GERMANY)

                                         INPUT DATA FOR CHANGES TO COMMON VARIABLES


NNL       0   3.000000E+00

NGAM      0   1.000000E+00

IWT       0   2.000000E+00

DOUSIN    0   1.000000E+00

IUSER     1   2.000000E+00

LAST      0  -1.000000E+00

END       0   0.000000E+00
NINTT     1
NSTEND  160    1.00000E+00     1.60000E+02
```



```
REFERENCE/RESPONSE - FUNCTION

 3.8892E+00  1.0428E+01  1.5728E+01  1.8743E+01  1.9630E+01  1.8948E+01  1.7288E+01  1.5136E+01  1.2841E+01  1.0627E+01
 8.6192E+00  6.8759E+00  5.4092E+00  4.2052E+00  3.2359E+00  2.4679E+00  1.8676E+00  1.4035E+00  1.0482E+00  7.7854E-01
 5.7536E-01  4.2328E-01  3.1012E-01  2.2635E-01  1.6463E-01  1.1936E-01  8.6283E-02  6.2201E-02  4.4726E-02  3.2084E-02
 2.2964E-02  1.6402E-02  1.1693E-02  8.3201E-03  5.9101E-03  4.1912E-03  2.9677E-03  2.0983E-03  1.4815E-03  1.0447E-03
 7.3573E-04  5.1752E-04  3.6362E-04  2.5521E-04  1.7894E-04  1.2534E-04  8.7708E-05  6.1321E-05  4.2835E-05  2.9897E-05
 2.0850E-05  1.4530E-05  1.0118E-05  7.0405E-06  4.8958E-06  3.4022E-06  2.3627E-06  1.6398E-06  1.1374E-06  7.8852E-07
 5.4633E-07  3.7832E-07  2.6184E-07  1.8113E-07  1.2524E-07  8.6555E-08  5.9791E-08  4.1284E-08  2.8494E-08  1.9658E-08
 1.3556E-08  9.3446E-09  6.4391E-09  4.4353E-09  3.0540E-09  2.1021E-09  1.4464E-09  9.9490E-10  6.8411E-10  4.7026E-10
 3.2315E-10  2.2200E-10  1.5246E-10  1.0467E-10  7.1846E-11  4.9300E-11  3.3820E-11  2.3194E-11  1.5903E-11  1.0901E-11
 7.4704E-12  5.1182E-12  3.5058E-12  2.4008E-12  1.6438E-12  1.1251E-12  7.7001E-13  5.2685E-13  3.6040E-13  2.4649E-13
 1.6855E-13  1.1523E-13  7.8762E-14  5.3826E-14  3.6778E-14  2.5125E-14  1.7161E-14  1.1719E-14  8.0019E-15  5.4627E-15
 3.7286E-15  2.5446E-15  1.7363E-15  1.1846E-15  8.0803E-16  5.5110E-16  3.7581E-16  2.5624E-16  1.7468E-16  1.1907E-16
 8.1151E-17  5.5300E-17  3.7679E-17  2.5669E-17  1.7485E-17  1.1909E-17  8.1101E-18  5.5223E-18  3.7598E-18  2.5595E-18
 1.7422E-18  1.1857E-18  8.0689E-19  5.4904E-19  3.7354E-19  2.5412E-19  1.7286E-19  1.1757E-19  7.9953E-20  5.4368E-20
 3.6966E-20  2.5132E-20  1.7085E-20  1.1613E-20  7.8928E-21  5.3640E-21  3.6450E-21  2.4767E-21  1.6827E-21  1.1431E-21
 7.7651E-22  5.2742E-22  3.5821E-22  2.4326E-22  1.6519E-22  1.1216E-22  7.6152E-23  5.1699E-23  3.5095E-23  2.3822E-23
```



```
                                     FINAL VALUES OF CONTROL VARIABLES
CONVRG = 5.00000E-05
PLMNMX =-5.09414E+03 5.09414E+03
PNMNMX = 2.00000E-02 2.08000E+00
RUSER  = 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
IRWCAP =           0
IUNIT  =           0
IWT    =           2
LINEPG =          60
MCONV  =           3
METHOD =           2
MIOERR =           5
NABORT =           3
ND     =           1
NERFIT =          10
NGAM   =           1
NNL    =           3
NZ     =          50
IFORMT = (5E15.6)
IFORMW = (5E15.6)
IFORMY = (5E15.6)
IPLFIT =           0           0
IPLRES =           0           1
IPRINT =           1           1           1           1           1           1
IPRITR =           0           2
IUSER  =           2           0           0           0           0           0           0           0           0           0
                   0           0           0           0           0           0           0           0           0           0
                   0           0           0           0           0           0           0           0           0           0
                   0           0           0           0           0           0           0           0           0           0
                   0           0           0           0           0           0           0           0           0           0
MTRY   =          10          20          20          20          20          20          20          20          20          20
                  20          50          50          50          50          50          50          50          50          50
MXITER =          20          40
NL     =           1           2           3           4           5           6           7           8           9          10
DOUSIN =           T
LAST   =           F
PRWT   =           T
PRY    =           T
SAMET  =           F
SIMULA =           F
DOADEX =           F           F           F           F           F           F           F           F           F           F
                   F           F           F           F           F           F           F           F           F           F
DOSPL  =           T           T           T           T           T           T           T           T           T           T
                   T           T           T           T           T           T           T           T           T           T
DOSTRT =           F           F           F           F           F           F           F           F           F           F
                   F           F           F           F           F           F           F           F           F           F
DOUSOU =           F           F
LUSER  =           F           F           F           F           F           F           F           F           F           F
                   F           F           F           F           F           F           F           F           F           F
                   F           F           F           F           F           F           F           F           F           F
```

DATA SET 1

T	Y	T	Y	T	Y	T	Y	T	Y
1.000E+00	5.80954E+02	2.000E+00	9.87631E+02	3.000E+00	1.94205E+03	4.000E+00	3.29751E+03	5.000E+00	4.72440E+03
6.000E+00	6.00509E+03	7.000E+00	7.05631E+03	8.000E+00	8.09584E+03	9.000E+00	8.93946E+03	1.000E+01	9.44329E+03
1.100E+01	9.72025E+03	1.200E+01	1.00739E+04	1.300E+01	9.90493E+03	1.400E+01	9.97284E+03	1.500E+01	9.78328E+03
1.600E+01	9.71406E+03	1.700E+01	9.56667E+03	1.800E+01	9.30556E+03	1.900E+01	9.09049E+03	2.000E+01	8.65068E+03
2.100E+01	8.39509E+03	2.200E+01	8.13499E+03	2.300E+01	8.05614E+03	2.400E+01	7.67957E+03	2.500E+01	7.37493E+03
2.600E+01	7.20880E+03	2.700E+01	7.02243E+03	2.800E+01	6.83888E+03	2.900E+01	6.54531E+03	3.000E+01	6.28211E+03
3.100E+01	6.11013E+03	3.200E+01	5.90699E+03	3.300E+01	5.75305E+03	3.400E+01	5.56679E+03	3.500E+01	5.37367E+03
3.600E+01	5.25530E+03	3.700E+01	5.14227E+03	3.800E+01	5.01866E+03	3.900E+01	4.93654E+03	4.000E+01	4.80799E+03
4.100E+01	4.66767E+03	4.200E+01	4.46559E+03	4.300E+01	4.30753E+03	4.400E+01	4.29396E+03	4.500E+01	4.36461E+03
4.600E+01	4.14842E+03	4.700E+01	4.01151E+03	4.800E+01	3.97279E+03	4.900E+01	3.89141E+03	5.000E+01	3.80125E+03
5.100E+01	3.67661E+03	5.200E+01	3.57186E+03	5.300E+01	3.59123E+03	5.400E+01	3.39443E+03	5.500E+01	3.31909E+03
5.600E+01	3.45877E+03	5.700E+01	3.22456E+03	5.800E+01	3.17825E+03	5.900E+01	3.15583E+03	6.000E+01	3.03296E+03
6.100E+01	3.05516E+03	6.200E+01	2.97074E+03	6.300E+01	2.80511E+03	6.400E+01	2.86016E+03	6.500E+01	2.81720E+03
6.600E+01	2.71919E+03	6.700E+01	2.68062E+03	6.800E+01	2.67764E+03	6.900E+01	2.66993E+03	7.000E+01	2.55919E+03
7.100E+01	2.38792E+03	7.200E+01	2.55743E+03	7.300E+01	2.41663E+03	7.400E+01	2.34405E+03	7.500E+01	2.26293E+03
7.600E+01	2.28579E+03	7.700E+01	2.22075E+03	7.800E+01	2.12548E+03	7.900E+01	2.18068E+03	8.000E+01	2.09437E+03
8.100E+01	2.10139E+03	8.200E+01	2.01835E+03	8.300E+01	1.98056E+03	8.400E+01	1.93850E+03	8.500E+01	1.99847E+03
8.600E+01	1.85521E+03	8.700E+01	1.93964E+03	8.800E+01	1.92697E+03	8.900E+01	1.83710E+03	9.000E+01	1.81483E+03
9.100E+01	1.76789E+03	9.200E+01	1.82283E+03	9.300E+01	1.73610E+03	9.400E+01	1.71855E+03	9.500E+01	1.71283E+03
9.600E+01	1.59648E+03	9.700E+01	1.63133E+03	9.800E+01	1.57267E+03	9.900E+01	1.58856E+03	1.000E+02	1.52747E+03
1.010E+02	1.48819E+03	1.020E+02	1.49406E+03	1.030E+02	1.37873E+03	1.040E+02	1.46651E+03	1.050E+02	1.51674E+03
1.060E+02	1.42373E+03	1.070E+02	1.33983E+03	1.080E+02	1.39672E+03	1.090E+02	1.40669E+03	1.100E+02	1.33749E+03
1.110E+02	1.27427E+03	1.120E+02	1.31390E+03	1.130E+02	1.26248E+03	1.140E+02	1.27558E+03	1.150E+02	1.25996E+03
1.160E+02	1.21056E+03	1.170E+02	1.27877E+03	1.180E+02	1.19586E+03	1.190E+02	1.23859E+03	1.200E+02	1.07492E+03
1.210E+02	1.18758E+03	1.220E+02	1.09693E+03	1.230E+02	1.15358E+03	1.240E+02	1.03315E+03	1.250E+02	1.14176E+03
1.260E+02	1.06751E+03	1.270E+02	1.02991E+03	1.280E+02	1.10927E+03	1.290E+02	1.01118E+03	1.300E+02	1.04475E+03
1.310E+02	1.04928E+03	1.320E+02	1.03831E+03	1.330E+02	9.30903E+02	1.340E+02	9.68112E+02	1.350E+02	9.34791E+02
1.360E+02	9.79705E+02	1.370E+02	9.52782E+02	1.380E+02	9.40582E+02	1.390E+02	9.56763E+02	1.400E+02	8.73679E+02
1.410E+02	8.73245E+02	1.420E+02	9.32861E+02	1.430E+02	9.06989E+02	1.440E+02	8.88141E+02	1.450E+02	8.65821E+02
1.460E+02	8.39173E+02	1.470E+02	8.55407E+02	1.480E+02	8.39656E+02	1.490E+02	8.50966E+02	1.500E+02	7.97723E+02
1.510E+02	7.84301E+02	1.520E+02	8.17942E+02	1.530E+02	8.06588E+02	1.540E+02	7.72983E+02	1.550E+02	7.97675E+02
1.560E+02	7.45588E+02	1.570E+02	7.84774E+02	1.580E+02	7.76142E+02	1.590E+02	7.56732E+02	1.600E+02	7.71707E+02

PRECIS = 1.86D-16   SRANGE = 1.00E+35   RANGE = 1.00D+35

BOUNDS FOR LINEAR PARAMETERS: -5.1318E+07 .LE. ALPHA, GAMMA .LE. 5.1318E+07
BOUNDS FOR NONLINEAR PARAMETERS: 1.2500E-04 .LE. LAMBDA .LE. 2.0800E+00

TEST DATA SET 1 *** CONVOLUTION *** PRELIMINARY UNWEIGHTED 1 COMPONENTS ANALYSIS

ERROR ANALYZ 3. (CHECK USERS GUIDE.) ***

ERROR ANALYZ 3. (CHECK USERS GUIDE.) ***

TEST DATA SET 1 *** CONVOLUTION *** BEST SOLUTION
IN PRELIMINARY UNWEIGHTED 1 COMPONENTS ANALYSIS

STANDARD DEVIATION OF FIT = STDFIT = 1.93129E+02

GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
9.2864E+02	+-	2.7164E+01	2.925	6.9094E+01	+-	4.7249E-01	0.684	3.5090E-02	+-	4.3227E-04	1.232

CORRELATION COEFFICIENTS

	GAM 1	ALP 1	LAM 1
GAM 1	1.000		
ALP 1	-0.043	1.000	
LAM 1	0.680	0.525	1.000

TEST DATA SET 1 *** CONVOLUTION *** BEST SOLUTION
IN PRELIMINARY UNWEIGHTED 2 COMPONENTS ANALYSIS

STANDARD DEVIATION OF FIT = STDFIT = 5.11715E+01

GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
4.0240E+02	+-	2.0689E+01	5.141	4.0990E+01	+-	9.0641E-01	2.211	7.9625E-02	+-	1.8625E-03	2.339
				4.0125E+01	+-	8.5836E-01	2.139	1.9968E-02	+-	4.1309E-04	2.069

CORRELATION COEFFICIENTS

	GAM 1	ALP 1	ALP 2	LAM 1	LAM 2
GAM 1	1.000				
ALP 1	-0.769	1.000			
ALP 2	0.612	-0.932	1.000		
LAM 1	0.404	-0.730	0.916	1.000	
LAM 2	0.868	-0.958	0.911	0.732	1.000

TEST DATA SET 1 *** CONVOLUTION *** PRELIMINARY UNWEIGHTED 3 COMPONENTS ANALYSIS

ERROR ANALYZ 2. (CHECK USERS GUIDE.) ***

TEST DATA SET 1 *** CONVOLUTION ***

BEST SOLUTION
IN PRELIMINARY UNWEIGHTED 3 COMPONENTS ANALYSIS

STANDARD DEVIATION OF FIT = STDFIT = 5.11563E+01

GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
4.3027E+02	+-	2.7736E+01	6.446	4.2590E+01	+-	2.7663E+00	6.495	2.0865E-02	+-	8.7329E-04	4.186
				3.4272E+02	+-	1.8174E+05	53030.086	1.0788E-01	+-	1.0326E+00	957.239
				-3.0520E+02	+-	1.8175E+05	59550.305	1.1181E-01	+-	1.1828E+00	1057.898

CORRELATION COEFFICIENTS

	GAM 1	ALP 1	ALP 2	ALP 3	LAM 1	LAM 2	LAM 3
GAM 1	1.000						
ALP 1	0.681	1.000					
ALP 2	0.311	0.841	1.000				
ALP 3	-0.311	-0.841	-1.000	1.000			
LAM 1	0.856	0.956	0.683	-0.683	1.000		
LAM 2	0.319	0.847	1.000	-1.000	0.690	1.000	
LAM 3	-0.303	-0.835	-1.000	1.000	-0.676	-1.000	1.000

FOUND BEST FIT TO DATA WITH 3 COMPONENTS.
PROBABILITIES THAT OTHER SOLUTIONS ARE WORSE:

PNG(1/ 3) = 1.000 PNG(2/ 3) = 0.646 PNG(3/ 3) = 0.000

PARAMETERS USED TO GENERATE WEIGHTS

GAMMA	ALPHA	LAMBDA
4.3027E+02	4.2590E+01	2.0865E-02
	3.4272E+02	1.0788E-01
	-3.0520E+02	1.1181E-01

ERRFIT FOR DATA SET 1 = 3.97E+01

SQUARE ROOTS OF LEAST SQUARES WEIGHTS

4.1308E-02	3.1676E-02	2.2363E-02	1.7459E-02	1.4657E-02	1.2937E-02	1.1832E-02	1.1103E-02	1.0622E-02	1.0310E-02
1.0119E-02	1.0017E-02	9.9827E-03	1.0000E-02	1.0057E-02	1.0146E-02	1.0260E-02	1.0393E-02	1.0542E-02	1.0702E-02
1.0872E-02	1.1050E-02	1.1233E-02	1.1419E-02	1.1609E-02	1.1801E-02	1.1995E-02	1.2189E-02	1.2383E-02	1.2578E-02
1.2772E-02	1.2965E-02	1.3158E-02	1.3350E-02	1.3541E-02	1.3731E-02	1.3920E-02	1.4108E-02	1.4294E-02	1.4480E-02
1.4665E-02	1.4849E-02	1.5032E-02	1.5214E-02	1.5396E-02	1.5577E-02	1.5757E-02	1.5936E-02	1.6115E-02	1.6293E-02
1.6471E-02	1.6649E-02	1.6826E-02	1.7003E-02	1.7180E-02	1.7357E-02	1.7534E-02	1.7710E-02	1.7887E-02	1.8064E-02
1.8240E-02	1.8417E-02	1.8594E-02	1.8772E-02	1.8949E-02	1.9127E-02	1.9305E-02	1.9484E-02	1.9663E-02	1.9842E-02
2.0021E-02	2.0201E-02	2.0382E-02	2.0563E-02	2.0744E-02	2.0926E-02	2.1108E-02	2.1291E-02	2.1474E-02	2.1658E-02
2.1842E-02	2.2027E-02	2.2212E-02	2.2398E-02	2.2584E-02	2.2771E-02	2.2958E-02	2.3146E-02	2.3334E-02	2.3522E-02
2.3711E-02	2.3901E-02	2.4090E-02	2.4280E-02	2.4471E-02	2.4662E-02	2.4853E-02	2.5045E-02	2.5236E-02	2.5429E-02
2.5621E-02	2.5814E-02	2.6007E-02	2.6200E-02	2.6393E-02	2.6586E-02	2.6780E-02	2.6974E-02	2.7167E-02	2.7361E-02
2.7555E-02	2.7749E-02	2.7943E-02	2.8137E-02	2.8331E-02	2.8524E-02	2.8718E-02	2.8911E-02	2.9104E-02	2.9297E-02
2.9490E-02	2.9683E-02	2.9875E-02	3.0067E-02	3.0259E-02	3.0450E-02	3.0641E-02	3.0831E-02	3.1021E-02	3.1210E-02
3.1399E-02	3.1588E-02	3.1776E-02	3.1963E-02	3.2149E-02	3.2335E-02	3.2521E-02	3.2705E-02	3.2889E-02	3.3072E-02
3.3254E-02	3.3436E-02	3.3617E-02	3.3796E-02	3.3975E-02	3.4153E-02	3.4330E-02	3.4506E-02	3.4681E-02	3.4855E-02
3.5028E-02	3.5200E-02	3.5371E-02	3.5541E-02	3.5710E-02	3.5877E-02	3.6044E-02	3.6209E-02	3.6373E-02	3.6536E-02

TEST DATA SET 1 *** CONVOLUTION ***

FINAL 1 COMPONENTS ANALYSIS

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	2.231D+05	0.00E+00	1.65E+03	5.04E-01	1.59E-04
7	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	2.225D+05	0.00E+00	1.31E+03	2.56E+00	4.21E-04
7	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	1.810D+05	0.00E+00	-1.35E+03	2.43E+01	2.94E-03
6	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	4.013D+04	0.00E+00	-7.51E+02	4.78E+01	1.26E-02
6	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	1.065D+05	0.00E+00	1.50E+03	1.23E+02	1.44E-01
6	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	1.409D+05	0.00E+00	1.55E+03	1.44E+02	2.34E-01
7	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	1.652D+05	0.00E+00	1.59E+03	1.70E+02	3.80E-01
6	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	1.818D+05	0.00E+00	1.61E+03	2.04E+02	6.17E-01
7	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	1.931D+05	0.00E+00	1.62E+03	2.44E+02	1.00E+00
8	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1
0	2.008D+05	0.00E+00	1.63E+03	2.83E+02	1.63E+00
8	1.998D+03	5.00E-01	7.62E+02	6.66E+01	3.15E-02

TEST DATA SET 1 *** CONVOLUTION ***

BEST SOLUTION
IN FINAL 1 COMPONENTS ANALYSIS

STANDARD DEVIATION OF FIT = STDFIT = 3.56736E+00

GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
7.6160E+02	+-	2.0175E+01	2.649	6.6592E+01	+-	6.6713E-01	1.002	3.1472E-02	+-	4.1866E-04	1.330

CORRELATION COEFFICIENTS

	GAM 1	ALP 1	LAM 1
GAM 1	1.000		
ALP 1	0.198	1.000	
LAM 1	0.727	0.653	1.000

PLOT OF WEIGHTED RESIDUALS FOR DATA SET 1 MAX = U = 8.5E+00 MIN = L =-1.3E+01 RANDOM RUNS PROB. = 0.0000

```
U--------**---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+
      * *  **
     * *                                                                                    *            *
             **                                                         *              ** *  *** * *         *       *
    *          ***                                                  ***  *  ** *****     * *    * * *** * * * * * **   * * * *  *
                  **                                    *    *  ****      **  *     * *                    *   * *  * *           *
0----------------------*-----------------------------------*--*--------*-------------------------------*------------------*---**-*-
                    ***                      *       *   ** *  *                                                        *   *
   *                    * ***                 * ***** **
                         *   *         ***  *  *
                              *********   **
  *



L*--------+---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+
                    20                  40                  60                  80                 100                 120

U---------+---------+---------+




 **
0---*-****--*------------------
   * *       *** * *
          **    * * * ** * ** *
                     *  * *  *





L---------+---------+---------+
                   150                 170                 190                 210                 230                 250
```


TEST DATA SET 1 *** CONVOLUTION ***

FINAL 2 COMPONENTS ANALYSIS

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	6.441D+04	0.00E+00	1.37E+03	-6.21E+02	2.03E-04	6.45E+02	5.37E-04
13	1.520D+02	4.59E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	6.138D+04	0.00E+00	1.36E+03	-1.68E+02	2.03E-04	1.93E+02	1.42E-03
14	1.520D+02	7.45E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	5.345D+04	0.00E+00	1.31E+03	-5.56E+01	2.03E-04	8.33E+01	3.75E-03
13	1.520D+02*	3.60E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	3.392D+04	0.00E+00	1.18E+03	-1.74E+01	2.03E-04	5.30E+01	9.92E-03
12	1.520D+02*	3.33E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	3.862D+03	0.00E+00	7.38E+02	-1.19E+00	2.03E-04	6.11E+01	2.62E-02
10	1.520D+02	4.19E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	3.461D+04	0.00E+00	-6.87E+01	8.13E+00	2.03E-04	9.92E+01	6.93E-02
8	1.520D+02	6.80E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.109D+05	0.00E+00	-6.22E+02	1.26E+01	2.03E-04	1.51E+02	1.83E-01
9	1.520D+02*	3.46E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.562D+05	0.00E+00	-1.05E+03	1.55E+01	2.03E-04	2.49E+02	4.84E-01
10	1.520D+02	2.05E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.756D+05	0.00E+00	-1.72E+03	1.94E+01	2.03E-04	4.72E+02	1.28E+00
11	1.520D+02	3.76E-03	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	6.028D+04	0.00E+00	1.36E+03	-2.40E+02	5.37E-04	2.65E+02	1.42E-03
13	1.520D+02	7.10E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	5.246D+04	0.00E+00	1.32E+03	-6.35E+01	5.37E-04	9.14E+01	3.75E-03
11	1.520D+02	4.51E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	3.325D+04	0.00E+00	1.20E+03	-1.88E+01	5.37E-04	5.45E+01	9.92E-03
14	1.520D+02	4.72E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	3.304D+04	0.00E+00	-2.16E+02	9.34E+00	5.37E-04	9.88E+01	6.93E-02
8	1.520D+02	2.87E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.061D+05	0.00E+00	-9.37E+02	1.50E+01	5.37E-04	1.52E+02	1.83E-01
8	1.520D+02*	3.37E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.492D+05	0.00E+00	-1.49E+03	1.87E+01	5.37E-04	2.58E+02	4.84E-01
9	1.520D+02*	2.63E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.663D+05	0.00E+00	-2.31E+03	2.37E+01	5.37E-04	5.05E+02	1.28E+00
8	1.520D+02*	3.60E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	4.986D+04	0.00E+00	1.34E+03	-9.57E+01	1.42E-03	1.24E+02	3.75E-03
10	1.520D+02	6.34E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	3.150D+04	0.00E+00	1.27E+03	-2.30E+01	1.42E-03	5.91E+01	9.92E-03
10	1.520D+02	4.58E-01	4.12E+02	4.04E+01	8.08E-02	4.07E+01	2.02E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	2.847D+04	0.00E+00	-5.42E+02	1.25E+01	1.42E-03	9.69E+01	6.93E-02
9	1.520D+02*	3.23E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	9.152D+04	0.00E+00	-1.62E+03	2.10E+01	1.42E-03	1.53E+02	1.83E-01
8	1.520D+02*	3.24E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.382D+05	0.00E+00	-3.47E+03	3.36E+01	1.42E-03	5.59E+02	1.28E+00
8	1.520D+02*	3.37E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	2.689D+04	0.00E+00	1.41E+03	-4.12E+01	3.75E-03	7.85E+01	9.92E-03
10	1.520D+02	6.80E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	5.375D+04	0.00E+00	-2.22E+03	3.19E+01	3.75E-03	1.39E+02	1.83E-01
7	1.520D+02*	3.16E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	7.412D+04	0.00E+00	-3.04E+03	3.91E+01	3.75E-03	2.56E+02	4.84E-01
8	1.520D+02*	2.84E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	7.587D+04	0.00E+00	-3.84E+03	4.58E+01	3.75E-03	5.29E+02	1.28E+00
9	1.520D+02	6.76E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	8.974D+04	0.00E+00	1.53E+03	3.32E+02	1.83E-01	-4.24E+02	4.84E-01
10	1.520D+02	4.32E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.408D+05	0.00E+00	1.64E+03	7.09E+02	4.84E-01	-1.07E+03	1.28E+00
10	1.520D+02*	3.23E-01	4.12E+02	4.07E+01	2.02E-02	4.04E+01	8.08E-02

TEST DATA SET 1 *** CONVOLUTION ***

BEST SOLUTION
IN FINAL 2 COMPONENTS ANALYSIS

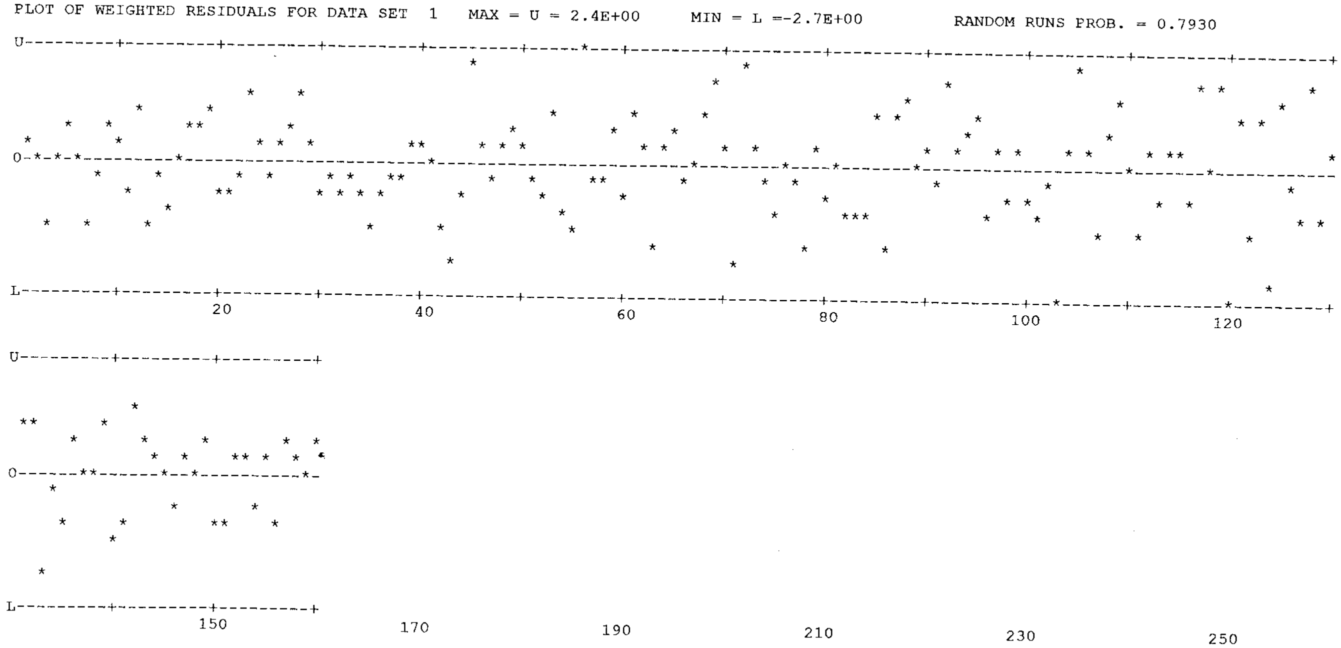
STANDARD DEVIATION OF FIT = STDFIT = 9.90131E-01

GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
4.1228E+02	+-	1.4287E+01	3.465	4.0436E+01	+-	8.6453E-01	2.138	8.0794E-02	+-	2.2196E-03	2.747
				4.0671E+01	+-	8.2201E-01	2.021	2.0220E-02	+-	3.3998E-04	1.681

CORRELATION COEFFICIENTS

	GAM 1	ALP 1	ALP 2	LAM 1	LAM 2
GAM 1	1.000				
ALP 1	-0.760	1.000			
ALP 2	0.655	-0.861	1.000		
LAM 1	0.418	-0.569	0.886	1.000	
LAM 2	0.876	-0.902	0.926	0.721	1.000

PLOT OF WEIGHTED RESIDUALS FOR DATA SET 1 MAX = U = 2.4E+00 MIN = L =-2.7E+00 RANDOM RUNS PROB. = 0.7930



TEST DATA SET 1 *** CONVOLUTION *** FINAL 3 COMPONENTS ANALYSIS

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.317D+04	0.00E+00	9.14E+02	1.80E+03	2.50E-04	-2.48E+03	1.00E-03	7.23E+02	4.02E-03
15	1.508D+02	7.92E-01	4.24E+02	4.41E+01	8.86E-02	-5.86E+00	1.87E-01	4.19E+01	2.06E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	6.832D+03	0.00E+00	7.87E+02	2.30E+02	2.50E-04	-2.65E+02	1.00E-03	9.21E+01	1.61E-02
26	1.508D+02	6.80E-01	4.24E+02	-5.87E+00	1.87E-01	4.41E+01	8.86E-02	4.19E+01	2.06E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.909D+03	0.00E+00	3.34E+02	-1.38E+02	2.50E-04	1.56E+02	1.00E-03	6.80E+01	6.46E-02
13	1.520D+02*	2.89E-03	4.12E+02	-9.87E+00	1.89E-02	5.04E+01	1.99E-02	4.05E+01	8.04E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	2.742D+04	0.00E+00	-1.39E+02	-2.24E+02	2.50E-04	2.53E+02	1.00E-03	1.15E+02	2.59E-01
40	1.520D+02	2.82E-03	4.13E+02	6.55E+01	1.94E-02	-2.52E+01	1.82E-02	4.07E+01	8.00E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	4.376D+04	0.00E+00	-5.95E+02	-2.41E+02	2.50E-04	2.75E+02	1.00E-03	2.59E+02	1.04E+00
16	1.508D+02	5.23E-01	4.24E+02	-5.88E+00	1.87E-01	4.19E+01	2.06E-02	4.41E+01	8.86E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	6.271D+03	0.00E+00	7.71E+02	4.12E+01	2.50E-04	-8.74E+01	4.02E-03	1.03E+02	1.61E-02
14	1.508D+02	4.81E-01	4.24E+02	4.41E+01	8.86E-02	-5.90E+00	1.87E-01	4.19E+01	2.06E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	2.216D+04	0.00E+00	-8.70E+01	-3.63E+01	2.50E-04	6.81E+01	4.02E-03	1.08E+02	2.59E-01
13	1.508D+02	5.07E-01	4.24E+02	-5.91E+00	1.87E-01	4.19E+01	2.06E-02	4.41E+01	8.86E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	3.616D+04	0.00E+00	-5.11E+02	-3.77E+01	2.50E-04	7.40E+01	4.02E-03	2.41E+02	1.04E+00
9	1.521D+02	6.76E-03	4.23E+02	4.21E+01	2.07E-02	-1.12E+02	6.93E-02	1.51E+02	7.26E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	5.887D+03	0.00E+00	1.27E+02	-1.69E+00	2.50E-04	4.61E+01	1.61E-02	7.23E+01	2.59E-01
8	1.519D+02	4.62E-03	4.13E+02	-2.06E+00	1.58E-02	4.25E+01	1.99E-02	4.06E+01	8.04E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.156D+05	0.00E+00	1.21E+03	2.24E+00	2.50E-04	3.33E+02	2.59E-01	-5.37E+02	1.04E+00
15	1.508D+02	3.56E-01	4.24E+02	4.19E+01	2.06E-02	4.41E+01	8.86E-02	-5.91E+00	1.87E-01

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	6.155D+03	0.00E+00	7.65E+02	5.80E+01	1.00E-03	-1.06E+02	4.02E-03	1.06E+02	1.61E-02
12	1.517D+02*	3.43E-03	4.18E+02	4.19E+01	2.05E-02	-1.06E+03	6.91E-02	1.10E+03	6.96E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.213D+03	0.00E+00	3.63E+02	-3.09E+01	1.00E-03	5.12E+01	4.02E-03	6.42E+01	6.46E-02
15	1.508D+02	5.00E-01	4.24E+02	-5.91E+00	1.87E-01	4.19E+01	2.06E-02	4.41E+01	8.86E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	2.123D+04	0.00E+00	-5.10E+01	-4.93E+01	1.00E-03	8.14E+01	4.02E-03	1.06E+02	2.59E-01
15	1.508D+02	5.77E-01	4.24E+02	-5.87E+00	1.87E-01	4.19E+01	2.06E-02	4.41E+01	8.86E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	3.488D+04	0.00E+00	-4.47E+02	-5.14E+01	1.00E-03	8.79E+01	4.02E-03	2.35E+02	1.04E+00
16	1.508D+02	5.35E-01	4.24E+02	-5.86E+00	1.87E-01	4.19E+01	2.06E-02	4.41E+01	8.86E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	5.793D+03	0.00E+00	1.82E+02	-2.30E+00	1.00E-03	4.66E+01	1.61E-02	7.11E+01	2.59E-01
8	1.519D+02	2.00E-02*	4.13E+02	-2.31E-01	1.30E-02	4.08E+01	2.01E-02	4.05E+01	8.06E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.131D+04	0.00E+00	-4.47E+01	-1.55E+00	1.00E-03	4.98E+01	1.61E-02	1.52E+02	1.04E+00
9	1.519D+02	4.01E-03	4.13E+02	-1.45E+00	1.64E-02	4.20E+01	2.00E-02	4.05E+01	8.07E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.102D+05	0.00E+00	-2.42E+02	1.16E+01	1.00E-03	2.88E+02	2.59E-01	-3.57E+02	1.04E+00
14	1.508D+02	6.40E-01	4.24E+02	4.19E+01	2.06E-02	4.42E+01	8.86E-02	-5.93E+00	1.86E-01

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	1.232D+04	0.00E+00	-5.14E+02	1.60E+01	4.02E-03	9.51E+01	6.46E-02	-3.05E+01	2.59E-01
15	1.508D+02	3.76E-01	4.24E+02	4.19E+01	2.06E-02	-5.92E+00	1.87E-01	4.41E+01	8.86E-02

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	5.950D+04	0.00E+00	-2.56E+03	3.59E+01	4.02E-03	1.60E+02	2.59E-01	1.89E+01	1.04E+00
13	1.508D+02	4.87E-01	4.24E+02	4.19E+01	2.06E-02	4.41E+01	8.86E-02	-5.91E+00	1.87E-01

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	2.335D+04	0.00E+00	1.24E+03	1.27E+02	6.46E-02	-1.25E+02	2.59E-01	7.10E+01	1.04E+00
11	1.508D+02	7.09E-01	4.24E+02	4.19E+01	2.06E-02	-5.85E+00	1.88E-01	4.41E+01	8.86E-02

TEST DATA SET 1 *** CONVOLUTION *** BEST SOLUTION IN FINAL 3 COMPONENTS ANALYSIS

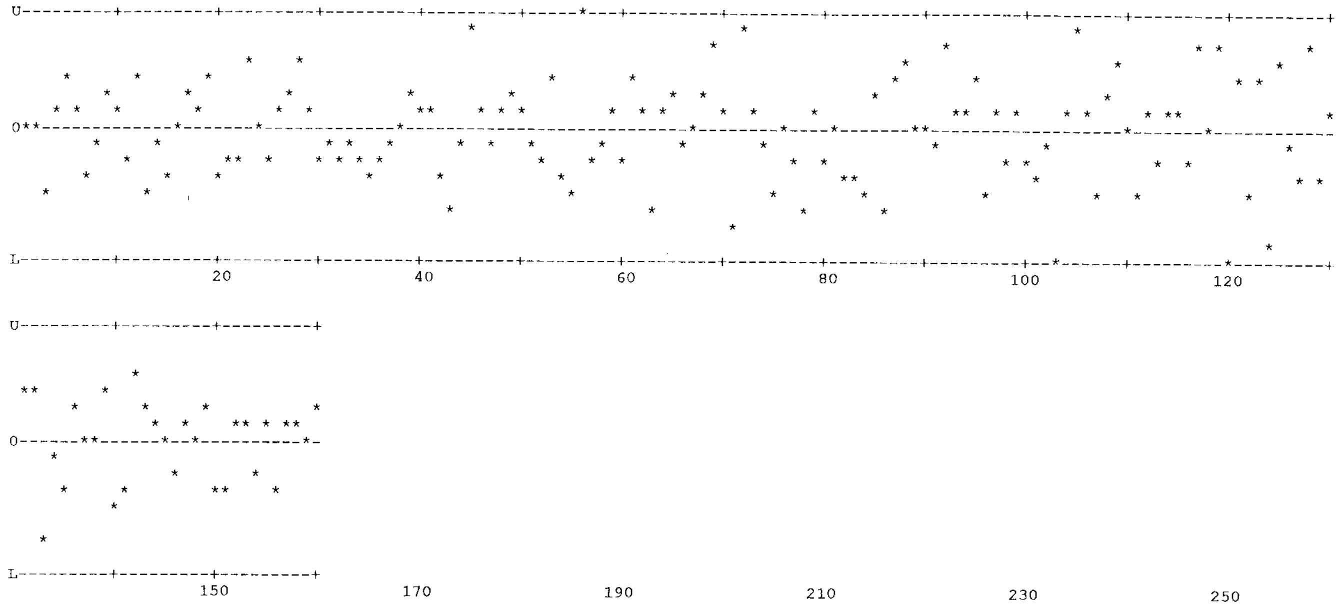
STANDARD DEVIATION OF FIT = STDFIT = 9.92791E-01

GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
4.2399E+02	+-	1.7469E+01	4.120	4.1895E+01	+-	1.5521E+00	3.705	2.0635E-02	+-	5.2173E-04	2.528
				4.4134E+01	+-	2.0019E+01	45.359	8.8623E-02	+-	1.7530E-02	19.780
				-5.9150E+00	+-	2.0504E+01	346.652	1.8678E-01	+-	3.1477E-01	168.530

CORRELATION COEFFICIENTS

	GAM 1	ALP 1	ALP 2	ALP 3	LAM 1	LAM 2	LAM 3
GAM 1	1.000						
ALP 1	0.702	1.000					
ALP 2	0.162	0.660	1.000				
ALP 3	-0.240	-0.726	-0.995	1.000			
LAM 1	0.877	0.951	0.477	-0.555	1.000		
LAM 2	0.363	0.840	0.958	-0.980	0.685	1.000	
LAM 3	-0.063	-0.548	-0.979	0.956	-0.363	-0.892	1.000

PLOT OF WEIGHTED RESIDUALS FOR DATA SET 1 MAX = U = 2.4E+00 MIN = L =-2.7E+00 RANDOM RUNS PROB. = 0.7726




```
FOUND BEST FIT TO DATA WITH    2   COMPONENTS.
PROBABILITIES THAT OTHER SOLUTIONS ARE WORSE:

PNG( 1/ 2) = 1.000      PNG( 2/ 2) = 0.000      PNG( 3/ 2) = 0.558
```



```
SPLMOD - VERSION 3DP (JUN 1988)          TEST DATA SET 2     ***  SUM OF EXPONENTIALS  ***

REFERENCES - S.W.PROVENCHER AND R.H.VOGEL (1983) IN PROGRESS IN SCIENTIFIC COMPUTING, VOL. 2, PAGES 304-319,
                                                 P.DEUFLHARD AND E.HAIRER EDS., (BIRKHAEUSER, BOSTON)
           - R.H.VOGEL                    (1988) SPLMOD USERS MANUAL, EMBL TECHNICAL REPORT DA09,
                                                 (EUROPEAN MOLECULAR BIOLOGY LABORATORY, HEIDELBERG, F.R. OF GERMANY)

                                    INPUT DATA FOR CHANGES TO COMMON VARIABLES


ND        0    3.000000E+00

NNL       0    2.000000E+00

NL        1    2.000000E+00

NL        2    3.000000E+00

NGAM      0    1.000000E+00

IWT       0    1.000000E+00

IFORMY    0    0.000000E+00
(4E20.6)

SAMET     0    1.000000E+00

DOUSOU    2    1.000000E+00

DOADEX    0    0.000000E+00
TTTTTTTTTTTTTTTTTT

IPRITR    2    3.000000E+00

IPLFIT    2    1.000000E+00

MTRY      0    0.000000E+00
  5    5    5    5    5    5    5    5    5    5    5    5    5    5    5    5    5    5    5    5

MXITER    2    2.000000E+01

IUSER     1    1.000000E+00

PLMNMX    1    0.000000E+00

PLMNMX    2    1.000000E+02

PNMNMX    1    2.000000E-01

DOUSIN    0   -1.000000E+00

LAST      0    1.000000E+00

END       0    0.000000E+00
NINTT     1
NSTEND   50     0.00000E+00     9.80000E+01
```

FINAL VALUES OF CONTROL VARIABLES

```
CONVRG = 5.00000E-05
PLMNMX = 0.00000E+00 1.00000E+02
PNMNMX = 2.00000E-01 2.08000E+00
RUSER  = 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
         0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
IRWCAP =          0
IUNIT  =          0
IWT    =          1
LINEPG =         60
MCONV  =          3
METHOD =          2
MIOERR =          5
NABORT =          3
ND     =          3
NERFIT =         10
NGAM   =          1
NNL    =          2
NZ     =         50
IFORMT = (5E15.6)
IFORMW = (5E15.6)
IFORMY = (4E20.6)
IPLFIT =          0      1
IPLRES =          0      1
IPRINT =          1      1      1      1      1      1
IPRITR =          0      3
IUSER  =          1      0      0      0      0      0      0      0      0      0
                  0      0      0      0      0      0      0      0      0      0
                  0      0      0      0      0      0      0      0      0      0
                  0      0      0      0      0      0      0      0      0      0
                  0      0      0      0      0      0      0      0      0      0
MTRY   =          5      5      5      5      5      5      5      5      5      5
                  5      5      5      5      5      5      5      5      5      5
MXITER =         20     20
NL     =          2      3      3      4      5      6      7      8      9     10
DOUSIN =          F
LAST   =          T
PRWT   =          T
PRY    =          T
SAMET  =          T
SIMULA =          F
DOADEX =          T      T      T      T      T      T      T      T      T      T
                  T      T      T      T      T      T      T      T      T      T
DOSPL  =          T      T      T      T      T      T      T      T      T      T
                  T      T      T      T      T      T      T      T      T      T
DOSTRT =          F      F      F      F      F      F      F      F      F      F
                  F      F      F      F      F      F      F      F      F      F
DOUSOU =          F      T
LUSER  =          F      F      F      F      F      F      F      F      F      F
                  F      F      F      F      F      F      F      F      F      F
                  F      F      F      F      F      F      F      F      F      F
```


DATA SET 1

T	Y	T	Y	T	Y	T	Y	T	Y
0.000E+00	3.92630E+01	2.000E+00	3.78621E+01	4.000E+00	3.41216E+01	6.000E+00	3.30735E+01	8.000E+00	3.15259E+01
1.000E+01	2.99061E+01	1.200E+01	2.91313E+01	1.400E+01	2.75977E+01	1.600E+01	2.82081E+01	1.800E+01	2.65212E+01
2.000E+01	2.57173E+01	2.200E+01	2.53186E+01	2.400E+01	2.49112E+01	2.600E+01	2.49183E+01	2.800E+01	2.34907E+01
3.000E+01	2.34972E+01	3.200E+01	2.35639E+01	3.400E+01	2.32833E+01	3.600E+01	2.28483E+01	3.800E+01	2.23709E+01
4.000E+01	2.20083E+01	4.200E+01	2.20761E+01	4.400E+01	2.17859E+01	4.600E+01	2.17283E+01	4.800E+01	2.14995E+01
5.000E+01	2.22927E+01	5.200E+01	2.07859E+01	5.400E+01	2.12752E+01	5.600E+01	2.07515E+01	5.800E+01	2.07152E+01
6.000E+01	2.06974E+01	6.200E+01	2.08856E+01	6.400E+01	2.00046E+01	6.600E+01	2.06331E+01	6.800E+01	2.09575E+01
7.000E+01	2.05997E+01	7.200E+01	2.10100E+01	7.400E+01	2.04239E+01	7.600E+01	1.98299E+01	7.800E+01	2.04870E+01
8.000E+01	2.01594E+01	8.200E+01	2.06341E+01	8.400E+01	2.03827E+01	8.600E+01	2.04851E+01	8.800E+01	2.04052E+01
9.000E+01	2.05013E+01	9.200E+01	1.98743E+01	9.400E+01	2.09660E+01	9.600E+01	2.01976E+01	9.800E+01	1.99987E+01

DATA SET 2

T	Y	T	Y	T	Y	T	Y	T	Y
0.000E+00	4.94667E+01	2.000E+00	4.45952E+01	4.000E+00	4.40464E+01	6.000E+00	4.11485E+01	8.000E+00	3.91268E+01
1.000E+01	3.69257E+01	1.200E+01	3.56110E+01	1.400E+01	3.34534E+01	1.600E+01	3.27403E+01	1.800E+01	3.16328E+01
2.000E+01	3.03412E+01	2.200E+01	2.93192E+01	2.400E+01	2.82534E+01	2.600E+01	2.79143E+01	2.800E+01	2.74521E+01
3.000E+01	2.62449E+01	3.200E+01	2.62892E+01	3.400E+01	2.57003E+01	3.600E+01	2.50714E+01	3.800E+01	2.45336E+01
4.000E+01	2.48013E+01	4.200E+01	2.35956E+01	4.400E+01	2.39378E+01	4.600E+01	2.35334E+01	4.800E+01	2.42268E+01
5.000E+01	2.11070E+01	5.200E+01	2.16601E+01	5.400E+01	2.23464E+01	5.600E+01	2.26617E+01	5.800E+01	2.24353E+01
6.000E+01	2.20892E+01	6.200E+01	2.15333E+01	6.400E+01	2.10741E+01	6.600E+01	2.09988E+01	6.800E+01	2.12725E+01
7.000E+01	2.17476E+01	7.200E+01	2.15578E+01	7.400E+01	2.03018E+01	7.600E+01	2.09608E+01	7.800E+01	2.06410E+01
8.000E+01	2.09450E+01	8.200E+01	1.98926E+01	8.400E+01	2.08796E+01	8.600E+01	1.99937E+01	8.800E+01	2.11655E+01
9.000E+01	2.03414E+01	9.200E+01	1.96332E+01	9.400E+01	2.04600E+01	9.600E+01	2.01782E+01	9.800E+01	1.99145E+01

DATA SET 3

T	Y	T	Y	T	Y	T	Y	T	Y
0.000E+00	2.99651E+01	2.000E+00	2.59961E+01	4.000E+00	2.16722E+01	6.000E+00	1.92823E+01	8.000E+00	1.65780E+01
1.000E+01	1.42475E+01	1.200E+01	1.19444E+01	1.400E+01	1.04221E+01	1.600E+01	8.63121E+00	1.800E+01	8.38118E+00
2.000E+01	7.30875E+00	2.200E+01	6.87681E+00	2.400E+01	5.80214E+00	2.600E+01	5.01164E+00	2.800E+01	4.84633E+00
3.000E+01	3.46130E+00	3.200E+01	3.75805E+00	3.400E+01	3.54401E+00	3.600E+01	3.23136E+00	3.800E+01	2.50307E+00
4.000E+01	2.59458E+00	4.200E+01	2.61215E+00	4.400E+01	1.97752E+00	4.600E+01	1.98894E+00	4.800E+01	1.54513E+00
5.000E+01	1.76231E+00	5.200E+01	1.07486E+00	5.400E+01	1.18563E+00	5.600E+01	8.90578E-01	5.800E+01	1.29350E+00
6.000E+01	7.44488E-01	6.200E+01	1.13555E+00	6.400E+01	7.00696E-01	6.600E+01	9.65748E-01	6.800E+01	9.31408E-01
7.000E+01	6.44300E-01	7.200E+01	8.94842E-01	7.400E+01	4.77920E-01	7.600E+01	-5.96124E-02	7.800E+01	2.25075E-01
8.000E+01	7.20417E-01	8.200E+01	6.48557E-01	8.400E+01	6.27091E-01	8.600E+01	3.99277E-01	8.800E+01	-1.06968E-01
9.000E+01	7.08857E-01	9.200E+01	6.11183E-01	9.400E+01	9.45263E-01	9.600E+01	1.65471E-01	9.800E+01	3.43607E-01

```
PRECIS = 1.86D-16          SRANGE = 1.00E+35     RANGE = 1.00D+35

BOUNDS FOR    LINEAR PARAMETERS:      0.0000E+00  .LE.  ALPHA, GAMMA  .LE.  4.9467E+03
BOUNDS FOR NONLINEAR PARAMETERS:      2.0000E-03  .LE.     LAMBDA     .LE.  1.0400E+00
```

TEST DATA SET 2 *** SUM OF EXPONENTIALS ***

FINAL 2 COMPONENTS ANALYSIS

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.739D+02	0.00E+00	1.96E+01	0.00E+00*	5.67E-03	1.77E+01	4.56E-02
			1.98E+01	0.00E+00*		2.78E+01	
			0.00E+00*	0.00E+00*		2.39E+01	
1	3.663D+01	1.36E+00	1.85E+01	2.93E+00	5.67E-03*	1.76E+01	6.45E-02
			1.29E+01	1.20E+01		2.43E+01	
			2.23E-01	0.00E+00*		2.80E+01	
2	3.534D+01	8.06E-02*	1.91E+01	2.49E+00	8.86E-03	1.74E+01	6.52E-02
			1.58E+01	9.68E+00		2.37E+01	
			2.54E-01	0.00E+00*		2.81E+01	
3	3.425D+01	8.00E-02*	1.93E+01	2.44E+00	1.12E-02	1.72E+01	6.57E-02
			1.68E+01	9.09E+00		2.32E+01	
			2.82E-01	0.00E+00*		2.81E+01	
4	3.333D+01	8.00E-02*	1.95E+01	2.48E+00	1.32E-02	1.71E+01	6.63E-02
			1.74E+01	8.95E+00		2.27E+01	
			3.08E-01	0.00E+00*		2.82E+01	
5	3.193D+01	1.46E-01*	1.96E+01	2.67E+00	1.64E-02	1.68E+01	6.72E-02
			1.81E+01	9.10E+00		2.19E+01	
			3.51E-01	0.00E+00*		2.83E+01	
6	2.850D+01	8.42E-01	2.01E+01	5.28E+00	3.50E-02	1.37E+01	7.22E-02
			1.97E+01	1.49E+01		1.41E+01	
			5.43E-01	1.50E-01		2.86E+01	
7	2.487D+01	1.08E+00	2.00E+01	9.94E+00	3.91E-02	9.46E+00	9.64E-02
			1.97E+01	2.07E+01		8.58E+00	
			1.34E-01	8.83E+00		2.11E+01	
8	2.482D+01	1.02E+00	1.99E+01	1.02E+01	3.90E-02	9.23E+00	9.97E-02
			1.97E+01	2.09E+01		8.38E+00	
			8.91E-02	9.49E+00		2.06E+01	
9	2.482D+01	1.03E+00	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.99E-02	9.53E+00		2.05E+01	
10	2.482D+01	1.00E+00*	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	
11	2.482D+01*	2.61E-01	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	1.202D+03	0.00E+00	3.99E+00	2.49E+01	5.67E-03	1.30E+01	3.67E-01
			0.00E+00*	3.32E+01		2.01E+01	
			0.00E+00*	5.34E+00		3.15E+01	
1	2.828D+02	1.00E+00*	1.24E+01	1.44E+01	7.67E-03	1.44E+01	1.49E-01
			5.76E+00	2.68E+01		1.92E+01	
			0.00E+00*	3.86E+00		3.09E+01	
2	2.502D+01	1.00E+00*	2.00E+01	1.05E+01	4.10E-02	8.87E+00	9.68E-02
			1.99E+01	2.17E+01		7.39E+00	
			1.78E-01	9.34E+00		2.05E+01	
3	2.482D+01	9.93E-01	1.99E+01	1.02E+01	3.90E-02	9.23E+00	9.98E-02
			1.97E+01	2.09E+01		8.39E+00	
			8.73E-02	9.51E+00		2.06E+01	
4	2.482D+01	9.89E-01	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	
5	2.482D+01	1.00E+00*	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	

6	2.482D+01	5.84E-01	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	

END OF FIT USING SPLINE APPROXIMATIONS, START OF FIT USING EXACT MODEL

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2
0	2.482D+01	0.00E+00	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	
1	2.482D+01*	1.25E-03	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	
2	2.482D+01	1.00E+00*	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	
3	2.482D+01	2.43E-02	1.99E+01	1.03E+01	3.91E-02	9.20E+00	9.98E-02
			1.97E+01	2.10E+01		8.33E+00	
			8.98E-02	9.53E+00		2.05E+01	

TEST DATA SET 2 *** SUM OF EXPONENTIALS ***

BEST SOLUTION
IN FINAL 2 COMPONENTS ANALYSIS

STANDARD DEVIATION OF FIT = STDFIT = 4.22574E-01

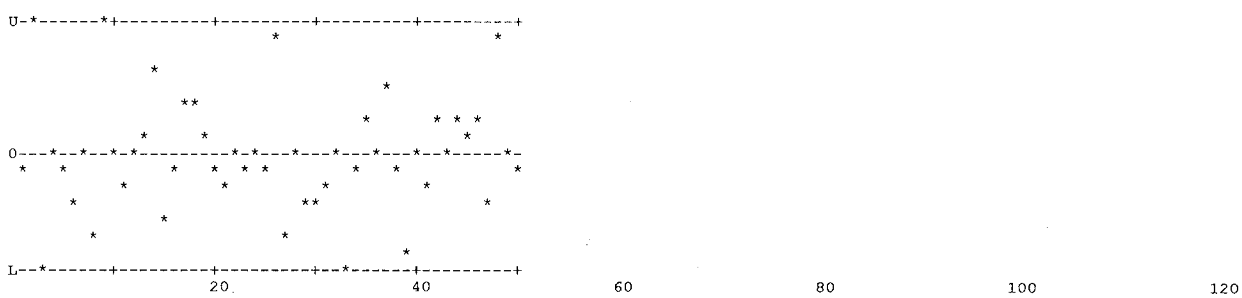
GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
1.9934E+01	+-	1.3844E-01	0.694	1.0258E+01	+-	1.6732E+00	16.312	3.9106E-02	+-	3.8789E-03	9.919
				9.2040E+00	+-	1.7653E+00	19.180	9.9782E-02	+-	9.1974E-03	9.218
1.9726E+01	+-	2.4198E-01	1.227	2.0987E+01	+-	2.4693E+00	11.766				
				8.3336E+00	+-	2.7811E+00	33.372				
8.9841E-02	+-	1.3415E-01	149.318	9.5318E+00	+-	2.5192E+00	26.430				
				2.0527E+01	+-	2.3891E+00	11.639				

ALPHA	1/(LAMBDA)	+-	STD. ERROR	PERCENT
1.0258E+01	2.5572E+01	+-	2.5365E+00	9.919
9.2040E+00	1.0022E+01	+-	9.2378E-01	9.218
2.0987E+01				
8.3336E+00				
9.5318E+00				
2.0527E+01				

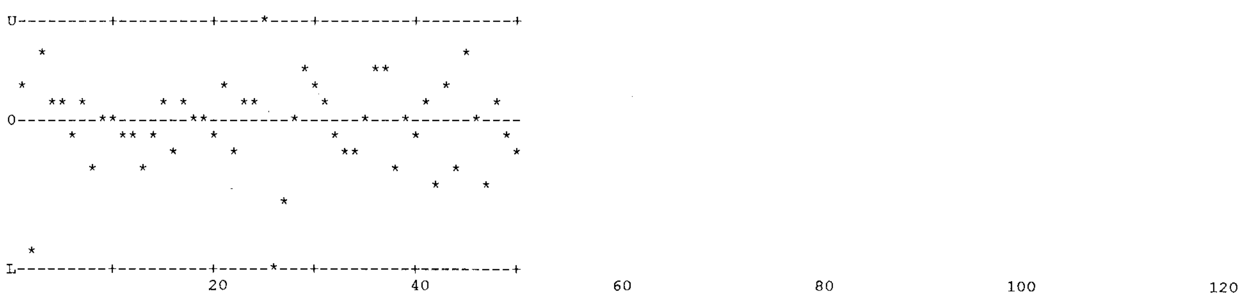
CORRELATION COEFFICIENTS

	GAM 1	ALP 1	ALP 2	GAM 1	ALP 1	ALP 2	GAM 1	ALP 1	ALP 2	LAM 1	LAM 2
GAM 1	1.000										
ALP 1	0.256	1.000									
ALP 2	-0.303	-0.985	1.000								
GAM 1	0.580	0.743	-0.758	1.000							
ALP 1	0.528	0.904	-0.894	0.747	1.000						
ALP 2	-0.547	-0.891	0.885	-0.774	-0.994	1.000					
GAM 1	0.286	-0.007	-0.041	0.272	0.084	-0.120	1.000				
ALP 1	0.389	0.902	-0.870	0.689	0.901	-0.879	-0.243	1.000			
ALP 2	-0.420	-0.896	0.869	-0.717	-0.905	0.887	0.192	-0.990	1.000		
LAM 1	0.610	0.881	-0.886	0.893	0.952	-0.958	0.215	0.844	-0.864	1.000	
LAM 2	0.266	0.866	-0.817	0.559	0.831	-0.796	-0.253	0.947	-0.912	0.728	1.000

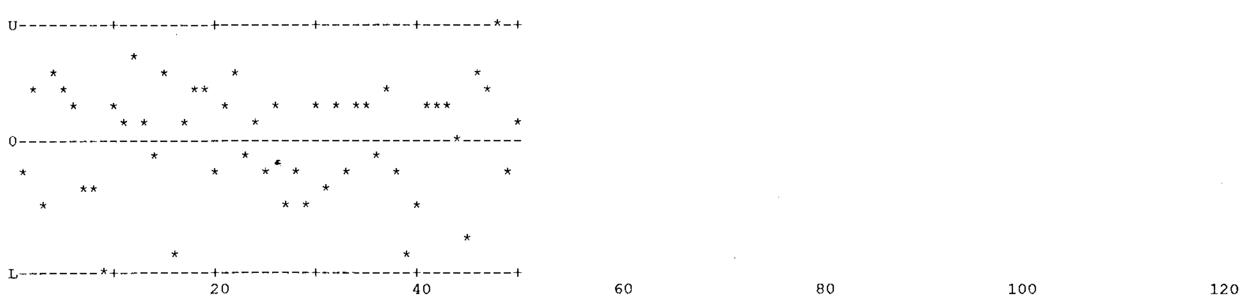
PLOT OF WEIGHTED RESIDUALS FOR DATA SET 1 MAX = U = 9.2E-01 MIN = L =-7.8E-01 RANDOM RUNS PROB. = 0.9438



PLOT OF WEIGHTED RESIDUALS FOR DATA SET 2 MAX = U = 1.2E+00 MIN = L =-1.6E+00 RANDOM RUNS PROB. = 0.9030



PLOT OF WEIGHTED RESIDUALS FOR DATA SET 3 MAX = U = 6.1E-01 MIN = L =-7.2E-01 RANDOM RUNS PROB. = 0.9207



PLOT OF DATA (O) AND FIT TO DATA (X) FOR DATA SET 1
ORDINATES LISTED ARE FIT VALUES.

ASSUMING 2 COMPONENTS

ORDINATE	ABSCISSA	PLOT
3.940E+01	0.000E+00	*
3.696E+01	2.000E+00	X O
3.488E+01	4.000E+00	O X
3.310E+01	6.000E+00	*
3.158E+01	8.000E+00	*
3.026E+01	1.000E+01	O X
2.913E+01	1.200E+01	*
2.814E+01	1.400E+01	O X
2.729E+01	1.600E+01	X O
2.654E+01	1.800E+01	OX
2.588E+01	2.000E+01	OX
2.530E+01	2.200E+01	*
2.479E+01	2.400E+01	XO
2.433E+01	2.600E+01	X O
2.393E+01	2.800E+01	O X
2.357E+01	3.000E+01	*
2.325E+01	3.200E+01	X O
2.296E+01	3.400E+01	X O
2.270E+01	3.600E+01	XO
2.246E+01	3.800E+01	*
2.225E+01	4.000E+01	OX
2.206E+01	4.200E+01	*
2.188E+01	4.400E+01	OX
2.172E+01	4.600E+01	*
2.158E+01	4.800E+01	*
2.145E+01	5.000E+01	X O
2.133E+01	5.200E+01	O X
2.122E+01	5.400E+01	*
2.112E+01	5.600E+01	O X
2.102E+01	5.800E+01	O X
2.094E+01	6.000E+01	O X
2.086E+01	6.200E+01	*
2.079E+01	6.400E+01	O X
2.072E+01	6.600E+01	*
2.066E+01	6.800E+01	X O
2.061E+01	7.000E+01	*
2.055E+01	7.200E+01	X O
2.051E+01	7.400E+01	*
2.046E+01	7.600E+01	O X
2.042E+01	7.800E+01	*
2.039E+01	8.000E+01	O X
2.035E+01	8.200E+01	X O
2.032E+01	8.400E+01	XO
2.029E+01	8.600E+01	XO
2.026E+01	8.800E+01	XO
2.024E+01	9.000E+01	XO
2.022E+01	9.200E+01	O X
2.019E+01	9.400E+01	X O
2.017E+01	9.600E+01	XO
2.016E+01	9.800E+01	OX

PLOT OF DATA (O) AND FIT TO DATA (X) FOR DATA SET 2
ORDINATES LISTED ARE FIT VALUES.

ASSUMING 2 COMPONENTS

ORDINATE	ABSCISSA	PLOT
4.905E+01	0.000E+00	XO
4.596E+01	2.000E+00	O X
4.327E+01	4.000E+00	X O
4.090E+01	6.000E+00	XO
3.883E+01	8.000E+00	XO
3.699E+01	1.000E+01	*
3.537E+01	1.200E+01	XO
3.393E+01	1.400E+01	OX
3.264E+01	1.600E+01	*
3.149E+01	1.800E+01	XO
3.046E+01	2.000E+01	OX
2.953E+01	2.200E+01	*
2.870E+01	2.400E+01	OX
2.794E+01	2.600E+01	OX
2.726E+01	2.800E+01	XO
2.664E+01	3.000E+01	O X
2.607E+01	3.200E+01	XO
2.556E+01	3.400E+01	*
2.509E+01	3.600E+01	*
2.466E+01	3.800E+01	OX
2.427E+01	4.000E+01	X O
2.391E+01	4.200E+01	OX
2.359E+01	4.400E+01	XO
2.328E+01	4.600E+01	XO
2.301E+01	4.800E+01	X O
2.275E+01	5.000E+01	O X
2.252E+01	5.200E+01	O X
2.230E+01	5.400E+01	*
2.211E+01	5.600E+01	X O
2.192E+01	5.800E+01	X O
2.176E+01	6.000E+01	XO
2.160E+01	6.200E+01	OX
2.146E+01	6.400E+01	OX
2.133E+01	6.600E+01	O X
2.120E+01	6.800E+01	*
2.109E+01	7.000E+01	X O
2.099E+01	7.200E+01	X O
2.089E+01	7.400E+01	O X
2.080E+01	7.600E+01	*
2.072E+01	7.800E+01	*
2.065E+01	8.000E+01	XO
2.058E+01	8.200E+01	O X
2.051E+01	8.400E+01	XO
2.045E+01	8.600E+01	OX
2.040E+01	8.800E+01	X O
2.035E+01	9.000E+01	*
2.030E+01	9.200E+01	O X
2.026E+01	9.400E+01	*
2.022E+01	9.600E+01	OX
2.018E+01	9.800E+01	*

PLOT OF DATA (O) AND FIT TO DATA (X) FOR DATA SET 3 ASSUMING 2 COMPONENTS
ORDINATES LISTED ARE FIT VALUES.

ORDINATE	ABSCISSA	
3.015E+01	0.000E+00	*
2.572E+01	2.000E+00	XO
2.201E+01	4.000E+00	OX
1.891E+01	6.000E+00	X O
1.630E+01	8.000E+00	XO
1.410E+01	1.000E+01	XO
1.225E+01	1.200E+01	OX
1.068E+01	1.400E+01	OX
9.347E+00	1.600E+01	O X
8.211E+00	1.800E+01	XO
7.240E+00	2.000E+01	*
6.407E+00	2.200E+01	XO
5.691E+00	2.400E+01	XO
5.071E+00	2.600E+01	*
4.535E+00	2.800E+01	XO
4.067E+00	3.000E+01	O X
3.660E+00	3.200E+01	*
3.302E+00	3.400E+01	XO
2.987E+00	3.600E+01	*
2.710E+00	3.800E+01	OX
2.464E+00	4.000E+01	*
2.245E+00	4.200E+01	XO
2.050E+00	4.400E+01	*
1.876E+00	4.600E+01	*
1.719E+00	4.800E+01	OX
1.579E+00	5.000E+01	*
1.452E+00	5.200E+01	OX
1.337E+00	5.400E+01	OX
1.234E+00	5.600E+01	OX
1.139E+00	5.800E+01	*
1.054E+00	6.000E+01	OX
9.758E-01	6.200E+01	XO
9.047E-01	6.400E+01	OX
8.397E-01	6.600E+01	*
7.803E-01	6.800E+01	*
7.259E-01	7.000E+01	*
6.761E-01	7.200E+01	XO
6.303E-01	7.400E+01	*
5.883E-01	7.600E+01	O X
5.497E-01	7.800E+01	OX
5.142E-01	8.000E+01	*
4.815E-01	8.200E+01	*
4.515E-01	8.400E+01	XO
4.238E-01	8.600E+01	*
3.982E-01	8.800E+01	OX
3.747E-01	9.000E+01	XO
3.530E-01	9.200E+01	XO
3.330E-01	9.400E+01	X O
3.145E-01	9.600E+01	OX
2.974E-01	9.800E+01	*

TEST DATA SET 2 *** SUM OF EXPONENTIALS ***

FINAL 3 COMPONENTS ANALYSIS

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	4.034D+01	0.00E+00	1.89E+01 1.75E+01 0.00E+00*	0.00E+00* 0.00E+00* 0.00E+00*	4.37E-03	7.65E+00 1.57E+01 4.85E+00	2.09E-02	1.33E+01 1.66E+01 2.62E+01	9.97E-02
1	2.506D+01	1.03E+00	1.98E+01 1.94E+01 7.49E-02	0.00E+00* 0.00E+00* 0.00E+00*	4.37E-03*	8.43E+00 1.81E+01 6.89E+00	3.42E-02	1.11E+01 1.16E+01 2.30E+01	9.16E-02
2	2.474D+01	1.33E+00	2.00E+01 1.81E+01 1.48E-01	0.00E+00* 2.62E+00 0.00E+00*	4.37E-03*	1.09E+01 2.05E+01 1.02E+01	4.15E-02	8.45E+00 7.91E+00 1.97E+01	1.01E-01
3	2.473D+01	7.32E-02	2.00E+01 1.57E+01 1.49E-01	0.00E+00* 4.94E+00 0.00E+00*	2.00E-03	1.10E+01 2.07E+01 1.04E+01	4.18E-02	8.33E+00 7.74E+00 1.96E+01	1.01E-01
4	2.468D+01	1.00E+00*	1.91E+01 1.26E+01 1.95E-01	1.19E+00 8.86E+00 0.00E+00*	2.00E-03*	1.24E+01 2.21E+01 1.30E+01	4.63E-02	6.69E+00 5.52E+00 1.70E+01	1.09E-01
5	2.468D+01	9.66E-01	1.91E+01 1.26E+01 1.88E-01	1.13E+00 8.85E+00 0.00E+00*	2.00E-03*	1.25E+01 2.22E+01 1.33E+01	4.65E-02	6.56E+00 5.38E+00 1.67E+01	1.11E-01
6	2.468D+01	9.07E-01	1.92E+01 1.26E+01 1.87E-01	1.09E+00 8.79E+00 0.00E+00*	2.00E-03*	1.25E+01 2.22E+01 1.33E+01	4.64E-02	6.58E+00 5.41E+00 1.68E+01	1.11E-01
7	2.468D+01	1.00E+00*	1.92E+01 1.26E+01 1.87E-01	1.09E+00 8.79E+00 0.00E+00*	2.00E-03*	1.25E+01 2.22E+01 1.33E+01	4.64E-02	6.58E+00 5.41E+00 1.68E+01	1.11E-01
8	2.468D+01*	1.64E-01	1.92E+01 1.26E+01 1.87E-01	1.09E+00 8.79E+00 0.00E+00*	2.00E-03*	1.25E+01 2.22E+01 1.33E+01	4.64E-02	6.58E+00 5.41E+00 1.68E+01	1.11E-01

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	5.928D+02	0.00E+00	1.70E+01 1.51E+01 0.00E+00*	0.00E+00* 0.00E+00* 0.00E+00*	4.37E-03	1.49E+01 2.49E+01 1.19E+01	2.09E-02	9.16E+00 1.10E+01 2.21E+01	4.76E-01
1	2.612D+01	1.21E+00	1.85E+01 1.38E+01 2.22E-01	2.52E+00 9.44E+00 0.00E+00*	4.37E-03*	1.59E+01 2.50E+01 2.23E+01	5.70E-02	2.64E+00 7.48E-01 8.04E+00	2.01E-01
2	2.472D+01	9.25E-01	1.99E+01 1.94E+01 2.22E-01	3.18E+00 8.31E+00 8.68E-01	2.65E-02	1.11E+01 1.74E+01 1.64E+01	5.50E-02	5.31E+00 3.89E+00 1.27E+01	1.23E-01
3	2.472D+01	8.00E-02*	1.99E+01 1.94E+01 2.36E-01	2.69E+00 7.41E+00 4.86E-01	2.53E-02	1.13E+01 1.80E+01 1.63E+01	5.34E-02	5.52E+00 4.18E+00 1.32E+01	1.21E-01
4	2.471D+01	8.00E-02*	1.99E+01 1.94E+01 2.55E-01	2.19E+00 6.49E+00 1.20E-01	2.39E-02	1.16E+01 1.87E+01 1.61E+01	5.19E-02	5.73E+00 4.46E+00 1.37E+01	1.20E-01
5	2.471D+01	2.00E-02*	1.99E+01 1.94E+01 2.60E-01	2.06E+00 6.25E+00 3.33E-02	2.35E-02	1.17E+01 1.89E+01 1.60E+01	5.15E-02	5.78E+00 4.53E+00 1.39E+01	1.19E-01
6	2.470D+01	1.00E-02*	1.99E+01 1.94E+01 2.62E-01	2.00E+00 6.13E+00 3.58E-10	2.33E-02	1.17E+01 1.90E+01 1.60E+01	5.13E-02	5.80E+00 4.56E+00 1.39E+01	1.19E-01
7	2.470D+01	2.23E-04	1.99E+01 1.94E+01 2.62E-01	2.00E+00 6.14E+00 3.58E-10*	2.33E-02	1.17E+01 1.90E+01 1.60E+01	5.13E-02	5.80E+00 4.56E+00 1.39E+01	1.19E-01
8	2.469D+01	1.34E+00	1.99E+01 1.91E+01 2.30E-01	1.15E+00 4.30E+00 3.58E-10*	1.66E-02	1.22E+01 2.06E+01 1.46E+01	4.89E-02	6.22E+00 5.01E+00 1.54E+01	1.14E-01

9	2.469D+01	1.45E+00	1.97E+01	9.50E-01	1.06E-02	1.27E+01	4.88E-02	6.01E+00	1.15E-01
			1.85E+01	4.06E+00		2.18E+01		4.67E+00	
			2.21E-01	3.58E-10*		1.47E+01		1.53E+01	
10	2.468D+01	1.75E+00	1.97E+01	6.58E-01	5.63E-03	1.23E+01	4.65E-02	6.67E+00	1.10E-01
			1.74E+01	4.25E+00		2.18E+01		5.54E+00	
			1.95E-01	3.58E-10*		1.32E+01		1.68E+01	
11	2.468D+01	7.28E-01	1.92E+01	1.09E+00	2.00E-03	1.25E+01	4.63E-02	6.61E+00	1.10E-01
			1.26E+01	8.76E+00		2.22E+01		5.44E+00	
			1.88E-01	3.58E-10*		1.32E+01		1.68E+01	
12	2.468D+01	9.59E-01	1.92E+01	1.10E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.79E+00		2.22E+01		5.40E+00	
			1.87E-01	3.58E-10*		1.33E+01		1.68E+01	
13	2.468D+01	1.00E+00*	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.79E+00		2.22E+01		5.41E+00	
			1.87E-01	3.58E-10*		1.33E+01		1.68E+01	
14	2.468D+01	4.92E-01	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.79E+00		2.22E+01		5.41E+00	
			1.87E-01	3.58E-10*		1.33E+01		1.68E+01	

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	9.387D+01	0.00E+00	1.03E+01	1.42E+01	4.37E-03	1.57E+01	9.97E-02	0.00E+00*	4.76E-01
			0.00E+00*	2.92E+01		2.17E+01		0.00E+00*	
			0.00E+00*	1.93E+00		3.02E+01		0.00E+00*	
1	2.972D+01	1.14E+00	1.44E+01	7.78E+00	3.41E-03	1.69E+01	7.07E-02	4.09E-01	4.76E-01*
			4.06E+00	2.17E+01		2.38E+01		0.00E+00*	
			0.00E+00*	7.03E-01		2.76E+01		1.86E+00	
2	2.896D+01	8.00E-02*	1.67E+01	5.65E+00	5.43E-03	1.67E+01	7.08E-02	4.90E-01	4.21E-01
			1.05E+01	1.56E+01		2.34E+01		0.00E+00*	
			0.00E+00*	8.12E-01		2.74E+01		2.03E+00	
3	2.835D+01	8.00E-02*	1.76E+01	4.89E+00	7.03E-03	1.65E+01	7.08E-02	5.68E-01	3.73E-01
			1.30E+01	1.33E+01		2.32E+01		0.00E+00*	
			0.00E+00*	9.11E-01		2.71E+01		2.22E+00	
4	2.783D+01	8.00E-02*	1.80E+01	4.51E+00	8.34E-03	1.63E+01	7.07E-02	6.49E-01	3.29E-01
			1.43E+01	1.21E+01		2.29E+01		0.00E+00*	
			0.00E+00*	1.00E+00		2.68E+01		2.44E+00	
5	2.737D+01	8.00E-02*	1.83E+01	4.29E+00	9.39E-03	1.61E+01	7.05E-02	7.37E-01	2.91E-01
			1.52E+01	1.14E+01		2.28E+01		0.00E+00*	
			0.00E+00*	1.07E+00		2.65E+01		2.71E+00	
6	2.697D+01	8.00E-02*	1.85E+01	4.14E+00	1.02E-02	1.60E+01	7.01E-02	8.35E-01	2.59E-01
			1.57E+01	1.10E+01		2.27E+01		0.00E+00*	
			0.00E+00*	1.13E+00		2.61E+01		3.04E+00	
7	2.598D+01	2.54E-01*	1.89E+01	3.82E+00	1.25E-02	1.54E+01	6.89E-02	1.24E+00	1.85E-01
			1.69E+01	9.91E+00		2.24E+01		0.00E+00*	
			0.00E+00*	1.33E+00		2.44E+01		4.43E+00	
8	2.505D+01	7.92E-01	1.95E+01	3.93E+00	2.00E-02	1.23E+01	6.56E-02	3.58E+00	1.14E-01
			1.88E+01	8.77E+00		2.13E+01		6.19E-02	
			0.00E+00*	2.36E+00		1.69E+01		1.08E+01	
9	2.470D+01	1.00E+00*	1.98E+01	1.44E+00	1.65E-02	1.22E+01	5.04E-02	5.97E+00	1.15E-01
			1.90E+01	4.76E+00		2.07E+01		4.59E+00	
			2.26E-01	1.29E-01		1.49E+01		1.49E+01	
10	2.470D+01	2.00E-02*	1.98E+01	1.36E+00	1.60E-02	1.22E+01	5.01E-02	6.01E+00	1.14E-01
			1.90E+01	4.62E+00		2.08E+01		4.65E+00	
			2.36E-01	6.58E-02		1.49E+01		1.50E+01	
11	2.470D+01	2.00E-02*	1.98E+01	1.28E+00	1.55E-02	1.23E+01	4.98E-02	6.05E+00	1.14E-01
			1.90E+01	4.48E+00		2.09E+01		4.71E+00	
			2.47E-01	3.80E-03		1.49E+01		1.51E+01	
12	2.470D+01	2.45E-03	1.98E+01	1.27E+00	1.55E-02	1.23E+01	4.97E-02	6.06E+00	1.14E-01
			1.90E+01	4.46E+00		2.09E+01		4.72E+00	
			2.48E-01	-1.86E-10		1.49E+01		1.51E+01	

13	2.470D+01*	3.14E-01	1.98E+01	1.57E+00	1.61E-02	1.27E+01	5.19E-02	5.38E+00	1.20E-01
			1.89E+01	5.06E+00		2.12E+01		3.84E+00	
			2.63E-01	2.78E-02		1.63E+01		1.36E+01	
14	2.470D+01	1.00E-02*	1.98E+01	1.53E+00	1.58E-02	1.27E+01	5.17E-02	5.41E+00	1.20E-01
			1.89E+01	4.98E+00		2.13E+01		3.88E+00	
			2.68E-01	-7.73E-10		1.63E+01		1.37E+01	
15	2.470D+01	2.45E-01	1.98E+01	1.50E+00	1.56E-02	1.28E+01	5.18E-02	5.31E+00	1.22E-01
			1.89E+01	4.95E+00		2.14E+01		3.78E+00	
			2.62E-01	-7.73E-10*		1.65E+01		1.35E+01	
16	2.469D+01	7.13E-01	1.98E+01	1.15E+00	1.42E-02	1.24E+01	4.94E-02	6.02E+00	1.15E-01
			1.89E+01	4.27E+00		2.12E+01		4.68E+00	
			2.37E-01	-7.73E-10*		1.49E+01		1.51E+01	
17	2.468D+01	1.79E+00	1.98E+01	7.31E-01	9.19E-03	1.25E+01	4.75E-02	6.35E+00	1.13E-01
			1.84E+01	3.76E+00		2.18E+01		5.14E+00	
			2.02E-01	-7.73E-10*		1.39E+01		1.61E+01	
18	2.468D+01	1.41E+00	1.94E+01	8.98E-01	3.52E-03	1.26E+01	4.69E-02	6.48E+00	1.11E-01
			1.57E+01	5.92E+00		2.22E+01		5.25E+00	
			1.95E-01	-7.73E-10*		1.35E+01		1.65E+01	
19	2.468D+01	2.61E-01	1.91E+01	1.19E+00	2.00E-03	1.25E+01	4.66E-02	6.54E+00	1.11E-01
			1.25E+01	8.96E+00		2.23E+01		5.34E+00	
			1.91E-01	-7.73E-10*		1.33E+01		1.67E+01	
20	2.468D+01	9.74E-01	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.59E+00	1.11E-01
			1.26E+01	8.78E+00		2.22E+01		5.41E+00	
			1.87E-01	-7.73E-10*		1.33E+01		1.68E+01	
21	2.468D+01	1.00E+00*	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.79E+00		2.22E+01		5.41E+00	
			1.87E-01	-7.73E-10*		1.33E+01		1.68E+01	
22	2.468D+01	4.35E-01	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.79E+00		2.22E+01		5.41E+00	
			1.87E-01	-7.73E-10*		1.33E+01		1.68E+01	

END OF FIT USING SPLINE APPROXIMATIONS, START OF FIT USING EXACT MODEL

ITR	VARIANCE	DAMPING Q	GAMMA 1	ALPHA 1	LAMBDA 1	ALPHA 2	LAMBDA 2	ALPHA 3	LAMBDA 3
0	2.468D+01	0.00E+00	1.92E+01	1.09E+00	2.00E-03	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.79E+00		2.22E+01		5.41E+00	
			1.87E-01	0.00E+00*		1.33E+01		1.68E+01	
1	2.468D+01*	1.00E+00*	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.78E+00		2.22E+01		5.41E+00	
			1.87E-01	0.00E+00*		1.33E+01		1.68E+01	
2	2.468D+01*	1.25E-03	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.78E+00		2.22E+01		5.41E+00	
			1.87E-01	0.00E+00*		1.33E+01		1.68E+01	
3	2.468D+01	6.23E-03	1.92E+01	1.09E+00	2.00E-03*	1.25E+01	4.64E-02	6.58E+00	1.11E-01
			1.26E+01	8.78E+00		2.22E+01		5.41E+00	
			1.87E-01	0.00E+00*		1.33E+01		1.68E+01	

TEST DATA SET 2 *** SUM OF EXPONENTIALS ***

BEST SOLUTION
IN FINAL 3 COMPONENTS ANALYSIS

STANDARD DEVIATION OF FIT = STDFIT = 4.27549E-01

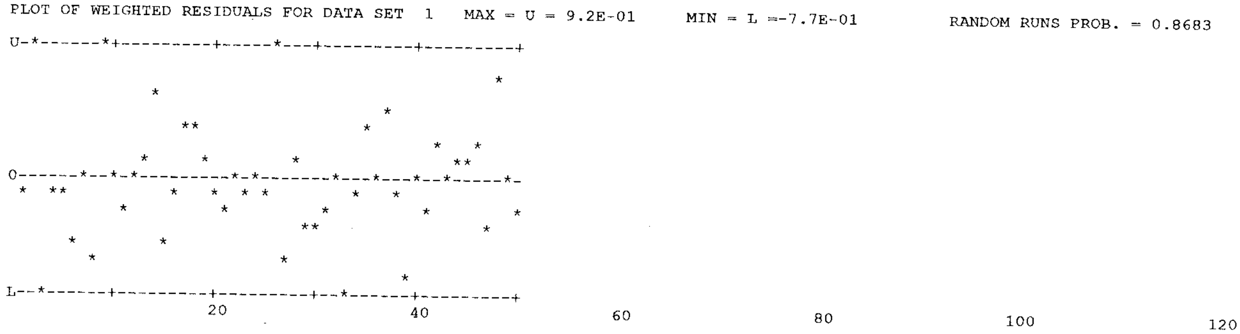
GAMMA	+-	STD. ERROR	PERCENT	ALPHA	+-	STD. ERROR	PERCENT	LAMBDA	+-	STD. ERROR	PERCENT
1.9169E+01	+-	6.0873E+01	317.553	1.0915E+00	+-	5.7013E+01	5223.387	2.0000E-03	+-	1.5689E-01	7844.690
				1.2520E+01	+-	5.3033E+00	42.357	4.6382E-02	+-	3.4436E-02	74.245
				6.5845E+00	+-	9.0864E+00	137.996	1.1056E-01	+-	4.1366E-02	37.416
1.2625E+01	+-	5.6236E+02	4454.354	8.7834E+00	+-	5.4796E+02	6238.610				
				2.2221E+01	+-	7.5928E+00	34.170				
				5.4095E+00	+-	9.7797E+00	180.789				
1.8697E-01	+-	1.0077E+01	5389.793	0.0000E+00	+-	1.2516E+01	**********				
				1.3265E+01	+-	1.2985E+01	97.892				
				1.6759E+01	+-	1.4704E+01	87.736				

ALPHA	1/(LAMBDA)	+-	STD. ERROR	PERCENT
1.0915E+00	5.0000E+02	+-	3.9223E+04	7844.690
1.2520E+01	2.1560E+01	+-	1.6007E+01	74.245
6.5845E+00	9.0451E+00	+-	3.3843E+00	37.416
8.7834E+00				
2.2221E+01				
5.4095E+00				
0.0000E+00				
1.3265E+01				
1.6759E+01				

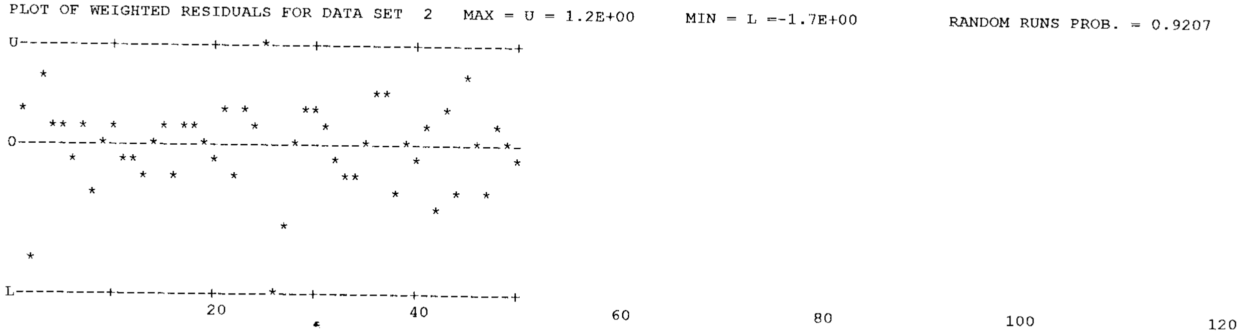
CORRELATION COEFFICIENTS

	GAM 1	ALP 1	ALP 2	ALP 3	GAM 1	ALP 1	ALP 2	ALP 3	GAM 1	ALP 1	ALP 2	ALP 3	LAM 1	LAM 2	LAM 3
GAM 1	1.000														
ALP 1	-1.000	1.000													
ALP 2	0.720	-0.710	1.000												
ALP 3	-0.855	0.844	-0.957	1.000											
GAM 1	0.997	-0.994	0.732	-0.875	1.000										
ALP 1	-0.997	0.994	-0.730	0.873	-1.000	1.000									
ALP 2	-0.891	0.897	-0.367	0.560	-0.874	0.875	1.000								
ALP 3	-0.759	0.743	-0.930	0.971	-0.794	0.791	0.415	1.000							
GAM 1	-0.803	0.793	-0.610	0.774	-0.828	0.827	0.702	0.750	1.000						
ALP 1	0.813	-0.802	0.625	-0.787	0.838	-0.837	-0.706	-0.764	-1.000	1.000					
ALP 2	0.803	-0.792	0.946	-0.968	0.825	-0.823	-0.481	-0.955	-0.629	0.646	1.000				
ALP 3	-0.846	0.834	-0.939	0.986	-0.869	0.867	0.544	0.970	0.717	-0.732	-0.992	1.000			
LAM 1	0.995	-0.992	0.741	-0.882	1.000	-1.000	-0.868	-0.803	-0.833	0.843	0.832	-0.876	1.000		
LAM 2	0.894	-0.881	0.886	-0.980	0.919	-0.917	-0.644	-0.961	-0.851	0.864	0.937	-0.972	0.926	1.000	
LAM 3	0.773	-0.762	0.952	-0.955	0.793	-0.791	-0.435	-0.944	-0.609	0.625	0.990	-0.977	0.800	0.912	1.000

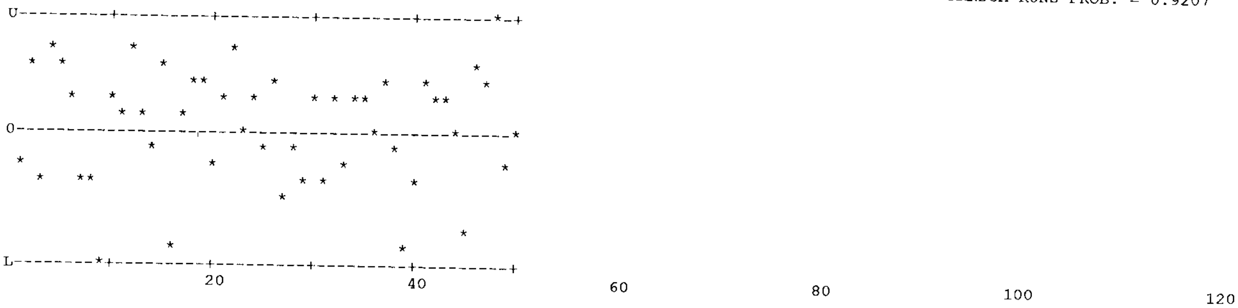
PLOT OF WEIGHTED RESIDUALS FOR DATA SET 1 MAX = U = 9.2E-01 MIN = L =-7.7E-01 RANDOM RUNS PROB. = 0.8683



PLOT OF WEIGHTED RESIDUALS FOR DATA SET 2 MAX = U = 1.2E+00 MIN = L =-1.7E+00 RANDOM RUNS PROB. = 0.9207

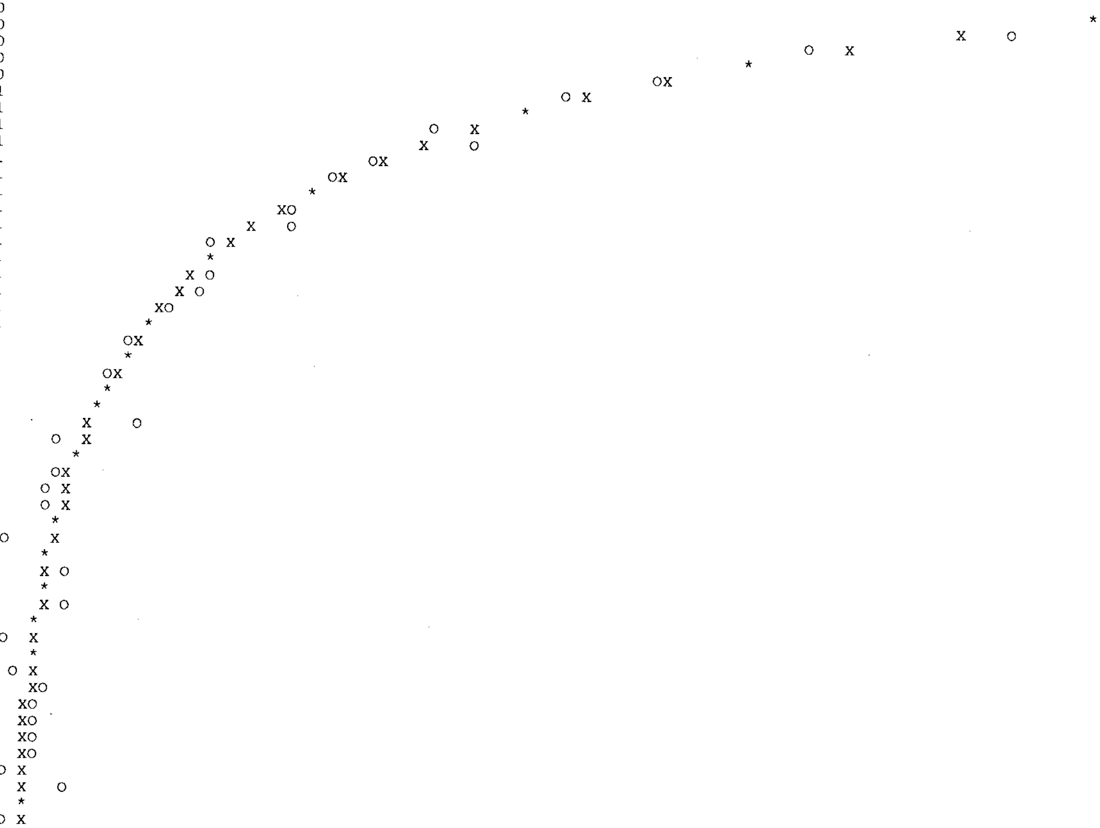


PLOT OF WEIGHTED RESIDUALS FOR DATA SET 3 MAX = U = 5.9E-01 MIN = L =-7.3E-01 RANDOM RUNS PROB. = 0.9207



PLOT OF DATA (O) AND FIT TO DATA (X) FOR DATA SET 1 ASSUMING 3 COMPONENTS
ORDINATES LISTED ARE FIT VALUES.

ORDINATE	ABSCISSA
3.937E+01	0.000E+00
3.695E+01	2.000E+00
3.488E+01	4.000E+00
3.312E+01	6.000E+00
3.160E+01	8.000E+00
3.029E+01	1.000E+01
2.916E+01	1.200E+01
2.817E+01	1.400E+01
2.731E+01	1.600E+01
2.656E+01	1.800E+01
2.589E+01	2.000E+01
2.531E+01	2.200E+01
2.479E+01	2.400E+01
2.433E+01	2.600E+01
2.392E+01	2.800E+01
2.355E+01	3.000E+01
2.322E+01	3.200E+01
2.293E+01	3.400E+01
2.267E+01	3.600E+01
2.243E+01	3.800E+01
2.221E+01	4.000E+01
2.202E+01	4.200E+01
2.185E+01	4.400E+01
2.169E+01	4.600E+01
2.154E+01	4.800E+01
2.141E+01	5.000E+01
2.130E+01	5.200E+01
2.119E+01	5.400E+01
2.109E+01	5.600E+01
2.100E+01	5.800E+01
2.092E+01	6.000E+01
2.085E+01	6.200E+01
2.078E+01	6.400E+01
2.072E+01	6.600E+01
2.066E+01	6.800E+01
2.061E+01	7.000E+01
2.056E+01	7.200E+01
2.052E+01	7.400E+01
2.048E+01	7.600E+01
2.044E+01	7.800E+01
2.041E+01	8.000E+01
2.038E+01	8.200E+01
2.035E+01	8.400E+01
2.032E+01	8.600E+01
2.030E+01	8.800E+01
2.027E+01	9.000E+01
2.025E+01	9.200E+01
2.023E+01	9.400E+01
2.022E+01	9.600E+01
2.020E+01	9.800E+01



```
PLOT OF DATA (O)  AND FIT TO DATA (X)  FOR DATA SET    2          ASSUMING   3   COMPONENTS
ORDINATES LISTED ARE FIT VALUES.

 ORDINATE   ABSCISSA
4.904E+01  0.000E+00   XO
4.596E+01  2.000E+00   O X
4.327E+01  4.000E+00   X O
4.091E+01  6.000E+00   *
3.884E+01  8.000E+00   XO
3.700E+01  1.000E+01   *
3.537E+01  1.200E+01   XO
3.392E+01  1.400E+01   OX
3.263E+01  1.600E+01   *
3.148E+01  1.800E+01   XO
3.044E+01  2.000E+01   OX
2.951E+01  2.200E+01   *
2.868E+01  2.400E+01   OX
2.792E+01  2.600E+01   OX
2.724E+01  2.800E+01   XO
2.662E+01  3.000E+01   O X
2.606E+01  3.200E+01   XO
2.555E+01  3.400E+01   *
2.508E+01  3.600E+01   *
2.466E+01  3.800E+01   OX
2.427E+01  4.000E+01   X O
2.392E+01  4.200E+01   OX
2.360E+01  4.400E+01   XO
2.330E+01  4.600E+01   XO
2.303E+01  4.800E+01   X   O
2.278E+01  5.000E+01   O     X
2.255E+01  5.200E+01   O  X
2.234E+01  5.400E+01   *
2.214E+01  5.600E+01   XO
2.196E+01  5.800E+01   X O
2.180E+01  6.000E+01   XO
2.164E+01  6.200E+01   OX
2.150E+01  6.400E+01   OX
2.137E+01  6.600E+01   O X
2.124E+01  6.800E+01   *
2.113E+01  7.000E+01   X O
2.102E+01  7.200E+01   XO
2.092E+01  7.400E+01   O X
2.083E+01  7.600E+01   *
2.074E+01  7.800E+01   *
2.065E+01  8.000E+01   XO
2.058E+01  8.200E+01O  X
2.050E+01  8.400E+01   XO
2.043E+01  8.600E+01 OX
2.037E+01  8.800E+01  X  O
2.030E+01  9.000E+01  *
2.024E+01  9.200E+01O X
2.019E+01  9.400E+01  *
2.013E+01  9.600E+01 *
2.008E+01  9.800E+01 *
```



```
PLOT OF DATA (O)  AND FIT TO DATA (X)  FOR DATA SET    3          ASSUMING   3   COMPONENTS
ORDINATES LISTED ARE FIT VALUES.

 ORDINATE   ABSCISSA
3.021E+01  0.000E+00   *
2.571E+01  2.000E+00   XO
2.198E+01  4.000E+00   OX
1.886E+01  6.000E+00   X O
1.626E+01  8.000E+00   XO
1.408E+01  1.000E+01   XO
1.224E+01  1.200E+01   OX
1.068E+01  1.400E+01   OX
9.360E+00  1.600E+01   O X
8.234E+00  1.800E+01   XO
7.269E+00  2.000E+01   *
6.440E+00  2.200E+01   XO
5.725E+00  2.400E+01   XO
5.105E+00  2.600E+01   *
4.565E+00  2.800E+01   XO
4.094E+00  3.000E+01   O X
3.681E+00  3.200E+01   *
3.318E+00  3.400E+01   XO
2.998E+00  3.600E+01   *
2.714E+00  3.800E+01   OX
2.463E+00  4.000E+01   *
2.239E+00  4.200E+01   XO
2.040E+00  4.400E+01   *
1.861E+00  4.600E+01   *
1.702E+00  4.800E+01   OX
1.558E+00  5.000E+01   XO
1.430E+00  5.200E+01   OX
1.314E+00  5.400E+01   OX
1.209E+00  5.600E+01   OX
1.115E+00  5.800E+01   *
1.030E+00  6.000E+01   OX
9.525E-01  6.200E+01   XO
8.827E-01  6.400E+01  OX
8.195E-01  6.600E+01   *
7.623E-01  6.800E+01   *
7.103E-01  7.000E+01  *
6.631E-01  7.200E+01  XO
6.203E-01  7.400E+01  *
5.814E-01  7.600E+01O X
5.460E-01  7.800E+01 OX
5.139E-01  8.000E+01  *
4.847E-01  8.200E+01  *
4.581E-01  8.400E+01  *
4.339E-01  8.600E+01 *
4.119E-01  8.800E+01OX
3.919E-01  9.000E+01 XO
3.736E-01  9.200E+01 XO
3.570E-01  9.400E+01 X O
3.419E-01  9.600E+01OX
3.281E-01  9.800E+01 *
```



```
FOUND BEST FIT TO DATA WITH    2   COMPONENTS.
PROBABILITIES THAT OTHER SOLUTIONS ARE WORSE:

PNG( 2/ 2) = 0.000       PNG( 3/ 2) = 0.940
```

4. MORE INFORMATION ON USING SPLMOD

If you are only interested in Applications you may be able to skip this section, or at least many parts of it (see Sec 6 for advice on what parts you should read). However, it is useful to at least look through this section to see what possibilities are available. (Remember that names of Control Variables, etc. are generally not cross-referenced and to use the indexes in Sec 8.)

4.1 USER Subprograms and more Control Variables

Section 4.1 contains descriptions of all the `USER` subprograms and all the Control Variables not described in Sec 3.4. The first four characters in the names of all the `USER` subprograms are `USER`; none of the Control Variables begin with `USER`. The `USER` subprograms are designed for you to change easily to suit your needs. They are quite well documented with comment cards; so the explanations in Sec 4.1 usually complement, rather than repeat, the information in these comment cards. The descriptions of the Control Variables here are in the same format as explained in Sec 3.4.

4.1.1 User arrays

You can use these Control Variables to input data that will be used in any of the `USER` subprograms or to store results that are produced in the `USER` subprograms for later use. Their use is illustrated in some of the `USER` subprograms described below. There is one array for `REAL`, `INTEGER`, and `LOGICAL` types:

`RUSER(100)`	
`IUSER(50)`	
`LUSER(30)`	This is a `LOGICAL` array.

These, like all the other Control Variables, are in three `COMMON` blocks (see the `BLOCK DATA` listing) that are shared by all the `USER` subprograms and many others. So be careful not to use the same array element (e.g., `IUSER(1)`) for two different purposes (even in different subprograms), since the first value will be overwritten.

If, for some reason, you ever want to change the `DIMENSION` specification of any of these three Control Variables, then you must not only change the `DIMENSION` specification in the `COMMON` block in **all** the subprograms, but also make the change explained in the comment cards in subprogram `STORIN` and change the corresponding `DATA` specification in `BLOCK DATA` so that the full array is initialized.

4.1.2 Analyzing N_D data sets simultaneously

SPLMOD has the possibility to analyze N_D datasets in parallel, each of which having the same time constants λ_j but with different amplitudes α_j and different γ_μ. In this case Eq (3.1–1) has to be rewritten

$$y_{kn} \approx \sum_{j=1}^{N_\lambda} \alpha_{jn} f(\lambda_j, t_{kn}) + \sum_{\mu=1}^{N_\gamma} \gamma_{\mu n} g_\mu(t_{kn}) =: \hat{y}(t_{kn}) \ , \quad \begin{array}{l} k = 1, \ldots, N_{y_n}, \\ n = 1, \ldots, N_D \end{array} \qquad (1)$$

and the variance to be minimized is now (see also Eq (3.2–1))

$$\mathtt{VAR} = \sum_{n=1}^{N_D} \sum_{k=1}^{N_{y_n}} w_{kn} (y_{kn} - \hat{y}_{kn})^2 \qquad (2)$$

For simplicity and convenience I have omitted the second subscript in all the previous sections. However to be able to explain all the facts in the following sections, from now on it is necessary to use the more general equations.

`ND`	N_D in Eq (2), i.e., the number of data sets to be analyzed simultaneously. (Default value is `ND=1`.)

The same conventions as described in Sec 3.3.2 are used to input the data y_{kn} and the t-values of Eq (1). The Card Sets 4–7 simply have to be repeated N_D times. However, with SPLMOD you have the possibility to avoid inputting the `T` array for every data set if the t-values are identical for all N_D data sets. You simply set the Control Variable `SAMET=.TRUE.` and input the `T` array for the first data set only, as described in Sec 3.3.2. Immediately after the data y_{k1} and possible weights w_{k1} of Data Set 1 are following the data y_{kn} and possible weights $w_{kn}, n = 2, \ldots, N_D$, for the next $N_D - 1$ data sets. This case is shown in Test Data Set 2 of File 3.

`SAMET = T`	to input the `T` array for all N_D data sets only once in Card 4a and Card Set 4b or in Card 5a and Card Set 5b, as described in Sec 3.3.2.
`= F`	to input the `T` array for every data set separately just before the corresponding `Y` array, as described in Sec 3.3.2. (This is the default value.)

The main advantage of **SAMET=T** is, that it helps to save a lot of computation time in avoiding the calculations of N_D times the same value of $f(\lambda_j, t_{kn})$ and $g_\mu(t_{kn})$ especially when $f(\lambda_j, t_{kn})$ is expensive to compute. Sec 4.1.7.1 tells you another possibility to save computation time.

4.1.3 Specifying the problem

4.1.3.1 Specifying USERTR

In SPLMOD all evaluations with respect to the nonlinear parameters λ are done on a transformed parameter axis, call it $H(\lambda)$.

USERTR	Fortran statements 110, 120, and 130 must define, respectively, the transformation $H(\lambda)$, the inverse transformation $H^{-1}(\lambda)$, and its derivative w.r.t. λ, $\partial H/\partial\lambda$.

The most common case of putting a parameter grid in equal logarithmic intervals is shown in the version of **USERTR** supplied in Files 1 and 2: $H(\lambda) = \ln(\lambda)$, $H^{-1}(\lambda) = \exp(\lambda)$, and $\partial H/\partial\lambda = 1/\lambda$. Note that it only makes sense to have $H(\lambda)$ a differentiable monotonic function of λ.

A logarithmic or similar transformation is often helpful (or essential) in providing a grid for an adequate representation of the solution, especially if λ ranges over several orders of magnitude. Usually, the choice of transformation is clear from the application, normally logarithmic or none.

4.1.3.2 Specifying the model function in Eq (4.1.2–1)

USERFN	evaluates $f(\lambda_j, t_{kn})$ in Eq (4.1.2–1) and its 1st and 2nd derivative.

The Fortran arguments of **USERFN** $= f(\lambda_j, t_{kn})$ in the notation of Eq (4.1.2–1) are **Z** $= H(\lambda_j)$, **J** $= j$, **NYMAX**, **T** $= t$ array, **NY** $= N_y$ array, **K** $= k$, **N** $= n$, **IDERIV**, and **RESPON**, with

IDERIV = 0 to obtain function value,
= 1 to obtain 1st derivative,
= 2 to obtain 2nd derivative,

and in case of a convolution **RESPON** contains the response function. **NYMAX** is only used in a **DIMENSION** statement for **T**.

The versions of **USERFN** in Files 1 and 2 illustrate the use of the user array **IUSER** and the use of recursion formulas; one recursion for a convolution (**IUSER(1)=2**), as in Test Data Set 1, and one for a pure exponential model (**IUSER(1)=1**), as in Test Data Set 2. A third possibility (**IUSER(1)=3**) is to use a reference instead of the time response of the instrument, as described in [6], but this will be explained in detail in Sec 6. In Sec 6 you will also find example programs, which do not use recursion formulas for exponentials or convolutions.

4.1.3.3 Specifying the N_γ terms in Eq (4.1.2–1)

NGAM	= N_γ	in the second sum in Eq (4.1.2–1).
	= 0	if there is no second sum in Eq (4.1.2–1), which is the default value.

NGAM must not be negative and must satisfy Eqs (3.5–2) and (3.5–3).

USERFL	(only called when **NGAM** is positive) evaluates the $g_\mu(t_{kn})$ in Eq (4.1.2–1).

The Fortran arguments of **USERFL** $= g_\mu(t_{kn})$, in the notation of Eq (4.1.2–1) are **MU** $= \mu$, **NYMAX**, **T** $= t$ array, **NY** $= N_y$ array, **K** $= k$, **N** $= n$, and **RESPON** which contains the response function in case of a convolution. **NYMAX** is only used in a **DIMENSION** statement for T.

The versions of **USERFL** in Files 1 and 2 illustrate the usual case when **NGAM** $= 1$, **USERFL** evaluates $g_1(t) = 1$; i.e., a simple additive constant (often called *background* or *baseline*) is provided for. However, when **NGAM** > 1 and one is dealing with a convolution, correction terms are provided for, which correct a shift of the response function [6]. For further details see Sec 6.

4.1.3.4 Specifying the least-squares weights

The w_{kn} in Eq (4.1.2–2) should be proportional to $1/\sigma_{kn}^2$, where σ_{kn} is the standard deviation of the noise at data point kn. If the noise level is independent of k, then **IWT=1** (an unweighted analysis) is appropriate. With **IWT=4**, you can input the w_{kn} directly. You can also use **IWT=1** and **USERIN** (see Sec 4.1.4) to compute the w_{kn} from the input data. If one of these cases apply, then you can skip the rest of Sec 4.1.3.4.

Often, however, σ_{kn} is a function of $\hat{y}(t_{kn})$, the (exact) value that y_{kn} would take if it were noise-free. For example, for *Poisson* statistics, $\sigma_{kn} = (\hat{y}(t_{kn}))^{1/2}$ and you should use $w_{kn} = 1/\hat{y}(t_{kn})$. Unfortunately, $\hat{y}(t_{kn})$ is unknown, and using y_{kn} in place of $\hat{y}(t_{kn})$ can dangerously bias the analysis, because y_{kn} that happen to have negative noise components (i.e., y_{kn} that are smaller than $\hat{y}(t_{kn})$) will be given too large w_{kn}, and y_{kn} with positive noise will be given too small w_{kn}. This is especially dangerous in the case of *Poisson* statistics and very small $\hat{y}(t_{kn})$, where y_{kn} could be nearly (or exactly) zero and w_{kn} could then become disastrously large.

Therefore SPLMOD has the option (with `IWT=2,3`, or `5`) of doing a **PRELIMINARY UNWEIGHTED ANALYSIS** to get better estimates of the $\hat{y}(t_{kn})$.These estimates are simply $\texttt{YFIT}_{kn}$, the fit to the y_{kn} (i.e., the $\hat{y}(t_{kn})$ computed from the solution) obtained in the **PRELIMINARY UNWEIGHTED ANALYSIS** (as on *Page 10* of the test output). Then, for example,

$$\texttt{YSAFE}_{kn} = \max\{\,|\texttt{YFIT}_{kn}|\,,\,\texttt{ERRFIT}_n\} \tag{1}$$

could be used in place of the unknown $\hat{y}(t_{kn})$ for computing the w_{kn}. $\texttt{ERRFIT}_n$ is an added safety margin to prevent a disastrously large w_{kn} if a $|\texttt{YFIT}_{kn}|$ happens to be very small and w_{kn} happens to be proportional to a negative power of $|\texttt{YFIT}_{kn}|$, as when `IWT=2` or `3`. It is computed by first finding the k for which $|\texttt{YFIT}_{kn}|$ is minimum and then setting $\texttt{ERRFIT}_n$ to the root-mean-square of the residuals, $y_{kn} - \texttt{YFIT}_{kn}$, for the `NERFIT` k centered at this minimum. Thus $\texttt{ERRFIT}_n$ is a rough estimate of the scatter of the y_{kn} near this minimum.

If there is no danger of w_{kn} blowing up for small $|\texttt{YFIT}_{kn}|$, then $\texttt{ERRFIT}_{kn}$ may not be necessary. In fact it may be possible to avoid the **PRELIMINARY UNWEIGHTED ANALYSIS** completely and to use the y_{kn} data directly to compute the w_{kn} (e.g., in `USERIN`), *provided* that the w_{kn} are insensitive to the noise in the y_{kn}.

If negative values of $\hat{y}(t_{kn})$ are meaningful, then the sign of $\texttt{YFIT}_{kn}$ could be appended to the right-hand side of Eq (1).

This **PRELIMINARY UNWEIGHTED ANALYSIS** strategy for determining least-squares weights has been widely used [e.g., 9,7,11], and *Price* [14] has pointed out that for *Poisson* statistics it yields approximately maximum-likelihood estimates. Although the computation time is doubled, we have mentioned in Sec 3.6.6 that it is useful to compare the results of the final weighted analysis with the preliminary unweighted one to see if the results are very sensitive to the weighting. You might even find evidence (e.g., in the residuals) that the unweighted analysis is more appropriate. There is much evidence that, when you

are uncertain whether a weighted or unweighted analysis is more appropriate, it is generally safer to favor the unweighted one.

`IWT = 1`	for $w_{kn} = 1$ for $k = 1, \ldots, N_{y_n}$, $n = 1, \ldots, N_D$ (i.e., an unweighted analysis, which is appropriate when σ_{kn} is independent of k).
`= 2`	for $w_{kn} = 1/\texttt{YSAFE}_{kn}$, where $\texttt{YSAFE}_{kn}$ is given by Eq (1). This is appropriate when σ_{kn} is proportional to $|\hat{y}(t_{kn})|^{1/2}$, as with *Poisson* statistics.
`= 3`	for $w_{kn} = [\texttt{YSAFE}_{kn}]^{-2}$. This is appropriate when σ_{kn} is proportional to $|\hat{y}(t_{kn})|$, i.e., for a constant *relative* error.
`= 4`	for inputting the w_{kn} in Card Set 7 (see Sec 3.3.2).
`= 5`	for computing the w_{kn} in `USERWT`. This is only necessary if none of the other `IWT` values are appropriate.

When `IWT=2`, `3`, or `5` a **PRELIMINARY UNWEIGHTED ANALYSIS**, as discussed above, is done. `IWT=3` weights small $\texttt{YSAFE}_{kn}$ very strongly; you should be certain that it is appropriate, and you should use $\texttt{ERRFIT}_n$.

`NERFIT` = n	the number of residuals used to compute $\texttt{ERRFIT}_n$, as discussed above. `NERFIT`=$O(10)$ is usually reasonable.
`= 0`	for setting $\texttt{ERRFIT}_n$=0.

`USERWT`	(only called when `IWT=5`) computes the w_{kn} after the **PRELIMINARY UNWEIGHTED ANALYSIS**.

In `USERWT` you are supplied with `NYMAX`, `Y` (the array of the y_{kn}), `T` (the array of the t_{kn}), `YLYFIT` (the array of the residuals, $y_{kn} - \texttt{YFIT}_{kn}$), `SQRTW` (the array of the w_{kn}, see below), `NY` array, `ERRFIT` array, and in case of a convolution `RESPON` contains the response function. `NYMAX` is only used in a `DIMENSION` statement for `Y`, `T`, `YLYFIT`, and `SQRTW`. From these you must compute and store the *square-roots* of the w_{kn} in `SQRTW(K,N)`, `K` $= 1, \ldots,$ `NY(N)`, `N` $= 1, \ldots,$ `ND`. You can evaluate $\texttt{YSAFE}_{kn}$ in Eq (1) as `AMAX1(ABS(Y(K,N)-YLYFIT(K,N)),ERRFIT(N))` with `K` $= k$ and `N` $= n$. You can then use $\texttt{YSAFE}_{kn}$ in place of the unknown $\hat{y}(t_{kn})$ to estimate the $1/\sigma_{kn}$ and store them in the `SQRTW(K,N)`.

In the version of `USERWT` in Files 1 and 2,

$$\sigma_{kn}^2 = \frac{\hat{y}^2(t_{kn}) + 1}{4B\hat{y}^2(t_{kn})} , \tag{2}$$

where B is a proportionality constant, independent of k, and can therefore be ignored when computing `SQRTW`. Also note that there is no danger of $1/\sigma_{kn}$ (and hence `SQRTW`) becoming disastrously large if the approximation to $\hat{y}(t_{kn})$ becomes very small; therefore `ERRFIT` is not necessary and `NERFIT=0` can be used. Equation (2) arises in photon correlation spectroscopy; it is derived in Eqs (9), (10), (13) and (14) of [2].

Equation (13) of [2] is a very useful general approximation for σ_{kn} when the y_{kn} have been obtained from a transformation of the original data, whose statistics are known.

4.1.4 Modifying the Input Data

`USERIN` provides a convenient way of inputting additional data needed, e.g., the response function in case of a convolution, as shown in Test Data Set 1, or for transforming your raw input data into the data required by SPLMOD, or for computing special least squares weights from the input data.

`DOUSIN = T`	to call `USERIN`
`= F`	to take no action

`USERIN`	(only called when `DOUSIN=T`) is called after the input of Card 1 – Card Set 7 (see Sec 3.3.2) and allows you to modify the t_{kn}, y_{kn}, and w_{kn} in Eqs (4.1.2–1) and (4.1.2–2) or the Control Variables.

In `USERIN` you are supplied with `NYMAX`, `Y` (the array of the y_{kn}), `T` (the array of the t_{kn}), `SQRTW` (the array of the w_{kn}, if any), `NY` array, and in case of a convolution `RESPON` contains the response function. `NYMAX` is only used in a `DIMENSION` statement for `Y`, `T`, and `SQRTW`. Note, that you put the w_{kn} (*not* their square roots, which are automatically computed by SPLMOD later) in `SQRTW`, when modifying the weights w_{kn} in `USERIN`, and that `IWT=4` has to be set after the modification.

`USERIN` is called before the output shown on *Page 2*. Thus changes that you have made in `USERIN` will be in this output.

In the versions of `USERIN` in Files 1 and 2, the response function for a convolution is simply read in and stored in array `RESPON`. This version will be used for Test Data Set 1. Additionally, it recomputes the Control Variable `PLMNMX`, i.e., the constraint for the linear parameters α_j, because in case of a convolution, the α_j depend on the spacing of the t_{kn} and the size of the response function. For choosing `PLMNMX` in other cases see Sec 3.4.3. The modified `PLMNMX` are printed together with `PNMNMX` on *Page 4* of the test output immediately after the data t_{kn} and y_{kn}, and during all the least squares analyses the α_j are constrained to be within these limits:

$$\texttt{PLMNMX}(1) \le \alpha_j \le \texttt{PLMNMX}(2) \quad , \qquad j = 1, \ldots, N_\lambda$$

In case of a convolution it is also necessary to have the background subtracted from the response function. The versions of `USERIN` in Files 1 and 2 provide you with an option to do this [i.e., if `LUSER(10)=T` subtract background stored in `RUSER(10)` from response function `RESPON`].

4.1.5 Output control

4.1.5.1 Special user-programmed output

`DOUSOU (2)`	`DOUSOU(ISTAGE) = T`	to call `USEROU`
	`= F`	to take no such action
	at Stage `ISTAGE`, where Stage 1 is the `PRELIMINARY UNWEIGHTED ANALYSIS` (as on *Pages 5–9* of the test output) and Stage 2 is the rest of the analysis (as on *Pages 11–21*).	

`USEROU`	(only called when `DOUSOU(ISTAGE)=T`) gives you the opportunity to use the solution and other terms described below to compute further output for your special purposes. It is described in the Fortran arguments list below (see `IROUTE`), when it is called.

As Fortran arguments in `USEROU`, you are supplied with `IROUTE`, which specifies from where the call to `USEROU` was and what can be used for doing your own output:

`IROUTE = 1` After generating your own starting values for the λ_j and before any analysis is done.

`IROUTE = 2` After every not necessarily successfull iteration.

`IROUTE = 3` After the plot of the solution for a single N_λ. It allows you, e.g., to output the parameters in a special form.

`IROUTE = 4` After the plot of the solution for a single N_λ and after a possible call to `USEROU` with `IROUTE=3`. It gives you the possibility to output the residuals or the fit to the data, e.g., by calling a special plot routine.

`IROUTE = 5` Has the same meaning as `IROUTE=3`, but is for the *best* solution only. This case is shown in Test Data Set 2 on *Pages 28* and *38*.

`IROUTE = 6` Has the same meaning as `IROUTE=4`, but is for the *best* solution only.

Further you are supplied with `NYMAX`, `Y` (the array of the y_{kn}), `T` (the array of the t_{kn}), `YLYFIT` (the array contains the residuals only when `DOSPL=F` (Sec 4.1.6) and `IROUTE>1` or when `DOSPL=T` and `IROUTE=4` or `6` *and* `IPLRES>0`), `SQRTW` (the array of the w_{kn}), `NY` (the array of the N_{y_n}), `NLINMX`, `NONLMX`, `PLIN` (the array of the linear parameters $\gamma_{\mu n}$, α_{jn}, for $\mu = 1,\ldots,N_\gamma$, $j = 1,\ldots,N_\lambda$, and $n = 1,\ldots,N_D$), `PLAM` (the array of the nonlinear parameters λ_j, for $j = 1,\ldots,N_\lambda$), `PLMHLP` (this array can be used as working space and has at least the same dimension as `PLAM`), `STDDEV` (has only a meaning if `IROUTE=3` or `5` and contains the diagonal elements of the inverted normal equations matrix, so it will allow you, together with `VAR` which is described below, e.g., to compute error estimates for the parameters), `VAR` (the variance of Eq (4.1.2–2), `ISTAGE`, and `RESPON` (the array contains the response function in case of a convolution). `NYMAX`, `NLINMX`, and `NONLMX` are used in `DIMENSION` statements.

The versions of `USEROU` in Files 1 and 2 illustrate the case `IROUTE=5`. They output the linear parameters α_{jn} and the reciprocal nonlinear parameters $1./\lambda_j$ of the best solution for a single N_λ together with their standard deviations. This output is only made in the final analysis (`ISTAGE=2`).

4.1.5.2 Controlling the quantity and spacing of the output

`MIOERR`	the number of messages indicating errors in your input format that will be printed before the run is aborted. `MIOERR=5` is reasonable. SPLMOD automatically sets `MIOERR` to at least 2.

`PRY`	`= T`	to output the `T` and `Y` arrays as on *Page 4* of the test output. If `ND>1`, both `T` and `Y` arrays are printed too, regardless of `SAMET` being `.TRUE.` or `.FALSE.` (see Sec 4.1.2 for `SAMET`). If `IWT=4`, the square roots of the w_{kn} in Eq (4.1.2–2) are also output. If `SIMULA=T`, then `EXACT`, the exact noise-free values, and the noise, `Y-EXACT`, used in the simulation are also printed.
	`= F`	to suppress this output. Unless N_y is excessively large, `PRY=T` is strongly recommended.

`PRWT`	`= T`	to output the `SQUARE ROOTS OF THE LEAST SQUARES WEIGHTS` (only when `IWT=2`, `3`, or `5`) as on *Page 10* of the test output.
	`= F`	to suppress this output. Unless N_y is excessively large, `PRWT=T` is strongly recommended.

`IPRITR` (2)		`IPRITR` (`ISTAGE`) controls at Stage `ISTAGE` the output of each iteration step, where Stage 1 is the `PRELIMINARY UNWEIGHTED ANALYSIS` (as on *Pages 5* and *8* of the test output) and Stage 2 is the `FINAL ANALYSIS` (as on *Pages 11*, *14*, *15*, *18*, and *19* of the test output).
	`= 0`	output no iterations at all (but, if errors occur a message is printed out as on *Pages 5* and *8* of the test output).
	`= 1`	output last iteration only for every try.
	`= 2`	output first and last iteration for every try. This will show you the starting values of the λ_j used in the optimization and their value after the last iteration step.
	`= 3`	output every iteration step. This will normally be a waste of paper, but is strongly recommended in suspicious cases or if any other difficulties arise. `IPRITR(1)=0` and `IPRITR(2)=2` are recommended.

`IPRINT` (3,2) `FORMAT(6I4)` `IPRINT(J,ISTAGE)=1,2, or, 3.`

The value of `IPRINT(J,ISTAGE)` controls the output of the solutions (see Sec 3.6.4.3) at Stage `ISTAGE`, where Stage 1 is the `PRELIMINARY UNWEIGHTED ANALYSIS` (as on *Pages 6, 7,* and *9* of the test output) and Stage 2 is the `FINAL ANALYSIS` on *Pages 12, 16,* and *20* of the test output).

`IPRINT(1,ISTAGE)` controls the output of the `STANDARD DEVIATION OF FIT = STDFIT` for a single N_λ:

- = 1 output `STDFIT` for the best solution only (this case is shown on *Pages 6, 7, 9, 12, 16,* and *20* of the test output).
- = 2 output `STDFIT` only when it is lower than one found in a previous solution.
- = 3 output `STDFIT` for every solution.

Only `IPRINT(1,ISTAGE)=1` is recommended.

`IPRINT(2,ISTAGE)` controls the output of the parameters `GAMMA`, `ALPHA`, and `LAMBDA` together with its standard errors `STD. ERROR` and the errors in percent `PERCENT`:

- = 1 output the above terms for the best solution only (as on *Pages 6, 7, 9, 12, 16,* and *20* of the test output).
- = 2 output the terms mentioned above only when `STDFIT` is lower than one found in a previous solution.
- = 3 output the terms mentioned above for every solution.

Only `IPRINT(2,ISTAGE)=1` is recommended.

`IPRINT(3,ISTAGE)` has exactly the same meaning and allowed values as `IPRINT(2,ISTAGE)`, except that it controls the output of the `CORRELATION COEFFICIENTS` (on *Pages 6, 7, 9, 12, 16,* and *20* of the test output is shown the case `IPRINT(3,ISTAGE)=1`).

Only `IPRINT(3,ISTAGE)=1` is recommended.

`IPLRES` (2) `IPLRES(ISTAGE)` controls at Stage `ISTAGE` when the weighted residuals (Sec 3.6.4.4) will be plotted (as on *Pages 13, 17, 21, 29–31,* and *39–41* of the test output).

- = 0 for never plotting them.
- = 1 for plotting them for the best solution only for a single N_λ (as on *Pages 13, 17,* and *21* in the test output).
- = 2 for also plotting them after every solution for a single N_λ with `STDFIT` lower than one found in a previous solution.
- = 3 for plotting them after every solution for a single N_λ.

`IPLRES(1)=0` and `IPLRES(2)=1` are recommended.

`IPLFIT` (2) `IPLFIT(ISTAGE)` has exactly the same meaning and allowed values as `IPLRES`, except that it controls when the plots of the data and the fit to the data (Sec 3.6.4.5) will be made.

Additionally negative arguments are allowed for `IPLFIT` too. It has been proven to be very usefull to plot sometimes only a part of the data and fit to the data (say from `KSTART` to `KEND`). To do this just enter `IPLFIT` with a negative argument and set `IUSER(19)` and `IUSER(20)` to `KSTART` and `KEND`, respectively. If you input a negative argument for `IPLFIT` and do not supply `IUSER(19)` and `IUSER(20)` or the latter do not take reasonable values, the whole plot will be printed.

`IPLFIT(1)=0` and `IPLFIT(2)=0` are recommended.

4.1.6 Remark on using Spline Approximations

As already described in Sec 3.2, SPLMOD usually approximates the nonlinear model function $f(\lambda, t)$ by cubic B-splines (Eq (4.1.2–1)), following the suggestions made in [1]

$$f(\lambda, t_{kn}) = \sum_{l=1}^{N_B} \beta_l(t_{kn}) B_l(\lambda) \tag{1}$$

where $B(\lambda)$ is a normalized cubic B-spline [13] defined on an equally spaced knot sequence $z_1 = H(\lambda_{\min}), \ldots, z_{N_B} = H(\lambda_{\max})$ (for $H(\cdot)$ see Sec 4.1.3.1), $N_B =$ `NZ` $+ 2$ and `NZ` is defined below. The two additional points are necessary because of continuity conditions at the end points of the knots' interval. $\lambda_{\min}$ and $\lambda_{\max}$ are defined by the constraints `PNMNMX(1)` and `PNMNMX(2)` respectively.

`NZ`	maximum number of equally spaced knots used for a B-spline approximation of the model function $f(\lambda, t)$ (Eq (4.1.2–1)). `NZ` must satisfy Eq (3.5–7).

Usually `NZ=50` is sufficient for a good approximation, but this will depend on $\lambda_{\min}$ and $\lambda_{\max}$ as shown above. If the range of allowed values for the λ_j is very large, it may be necessary to increase `NZ`.

If spline approximations have to be used or not is specified by Control Variable `DOSPL`.

`DOSPL` (10,2)	`FORMAT(20L1)` `DOSPL(J,ISTAGE)` specifies if the model function $f(\lambda, t)$ (Eq (4.1.2–1)) has to be approximated by B-splines or not for an analysis with $N_\lambda =$ `NL(J)` at Stage `ISTAGE`, where Stage 1 is the `PRELIMINARY UNWEIGHTED ANALYSIS` and Stage 2 is the `FINAL ANALYSIS`.
= `T`	if B-spline approximations are used.
= `F`	if exact model function is used.

The extra costs when using `DOSPL=T` are a few additional arrays to store the dot products of the B-spline coefficients $\beta(t_{kn})$ with $f(\lambda, t_{kn})$, $g(t_{kn})$, and y_{kn} respectively [1]. The additional amount of computing time to do this initial work is payed after a few iterations, especially if N_y is large, or if $f(\lambda, t)$ is a complicated function, e.g., a convolution, therefore `DOSPL=T` is recommended.

When you are analyzing several data sets in series and using `DOSPL=T` then you also have the possibility, under certain conditions which will be described in detail in Sec 4.1.7.1, to store the B-spline coefficients $\beta(t_{kn})$ and/or the dot products mentioned above from previous calculations and to use them in the following calculations.

There is also the possibility of doing a complete analysis for a single N_λ with `DOSPL=T`, and starting an analysis using the exact model function, i.e., `DOSPL=F`, after convergence is reached, thereby using the λ_j of the best solution as starting values. This is accomplished by the Control Variable `DOADEX`.

`DOADEX` (10,2)	`FORMAT(20L1)` `DOADEX(J,ISTAGE)` specifies if an additional analysis has to be performed, using the exact model function $f(\lambda, t)$ (Eq (4.2.1–1)), after a complete analysis with `DOSPL=T` for a single $N_\lambda =$ `NL(J)` at Stage `ISTAGE`, thereby using the λ_j of the best solution as starting values.
= `T`	to do an additional analysis using the exact model.
= `F`	to take no such action.

`DOADEX=T` may be useful, when you are assuming that the solution is biased because of rounding errors due to the B-spline approximations. Normally it is a waste of computer time to use `DOADEX=T` throughout all analyses, therefore only `DOADEX=F` is recommended.

4.1.7 Miscellaneous options

4.1.7.1 Scratch file — to save computing time

This is only of interest when using spline approximations, i.e., `DOSPL=T` (see Sec 4.1.6).

There are two possibilities of saving computer time. First you can store the B-spline coefficients $\beta_l(t_{kn})$ (Sec 4.1.6). This is especially of interest, if the computation in `USERFN` of $f(\lambda, t_{kn})$ in Eq (4.1.2–1) is time-consuming. The B-spline coefficients will then be computed only once and then written onto a temporary scratch file for later use. Second, if you have enough disk space, you can additionally store the dot products mentioned in Sec 4.1.6. The latter is only appliable when using `IWT=1` or `4`.

IRWCAP	= -2, -1, 0, 1, 2; specifies which quantities have to be stored for later use on a scratch file. The choice of IRWCAP will depend on IWT. The allowed combinations of IRWCAP and IWT and their functions are listed below.
	For IWT=1 or 4, it only makes sense to store any of the quantities described above, if you want to use them for analyzing other data sets with the same t_{kn}. You can save a remarkable amount of computer time, e.g., if you want to analyze a series of measurements which are all on the same time scale. Then the B-spline coefficients and the dot products have only to be computed in the analysis of the first data set, stored on a scratch file, and can then be used in the analyses of the rest of the data sets.
IRWCAP = -2	store B-spline coefficients and dot products for later use.
= -1	store dot products only.
= 0	do nothing.
= 1	read dot products from previous calculations (B-spline coefficients have to be computed).
= 2	read B-spline coefficients and dot products from previous calculations.
	For IWT=2, 3, or 5, you only have the possibility to store the B-spline coefficients, because the dot products mentioned in Sec 4.1.6 depend on the weights w_{kn} and the weights depend on the data y_{kn}, so they will be different for every data set.
IRWCAP = -2	do nothing.
= -1	store B-spline coefficients for later use.
= 0	do nothing.
= 1	read B-spline coefficients from previous calculations.
= 2	do nothing.

Even when you are analyzing one data set only, you can save computer time, especially if the computation of $f(\lambda, t_{kn})$ in USERFN is very time-consuming, by setting IRWCAP=-1. The B-spline coefficients calculated in the PRELIMINARY ANALYSIS will then be stored on a scratch file for later use in the FINAL ANALYSIS.

IUNIT	is the device number of the scratch file; e.g., $\beta_l(t_{kn})$ will be written and/or read WRITE(IUNIT)... and READ(IUNIT)..., respectively.

If you use a scratch file, the comment cards in Lines MAIN0159–MAIN0162 of the main subprogram point out that you may have to open the scratch file numbered IUNIT in the line MAIN0185 if your system does not do this automatically for you. This scratch file must be of the type that will allow REWINDs, unformatted WRITEs, and READs (iterated in that order) to be executed (in subprogram INICAP). The file requires enough space for the $N_B N_y N_D$ B-spline coefficients and, depending on IRWCAP described above, for $((N_B + N_\gamma + 1)N_B + (N_\gamma + 1)N_\gamma)N_D$ numbers (DOUBLE PRECISION for Version 3DP and REAL for Version 3SP), where N_γ, N_y, N_D, and N_B are defined in Secs 3.1, 4.1.2, and 4.1.6 respectively.

4.1.7.2 Simulating data

SIMULA = T	to call USERSI and USEREX to compute simulated data, y_{kn}, and put them in the array Y.
= F	to input the array Y in Card Set 6 (Sec 3.3.2).

USERSI	(only called when SIMULA=T) computes the simulated data, y_{kn}.

As a Fortran argument in USERSI, you are supplied with NYMAX, Y (the array of the simulated noisy data y_{kn}, see below), T (the array of the t_{kn} in Eq (4.2.1–1)), EXACT (the array of the simulated noise-free data, see below), NY (the array of the N_{y_n}), and in case of a convolution RESPON may contain the response function. NYMAX is only used in a DIMENSION statement for Y, T, and EXACT. In USERSI you must use USEREX to put the noise-free simulated *data* in the array EXACT(K,N) for K=1,...,NY(N), N=1,...,ND. You then add pseudo-random normal deviates (furnished by the subprogram RGAUSS) to EXACT, and store this simulated (noisy) data in Y(K,N) for K=1,...,NY(N), N=1,...,ND. You must have set IUSER(3) to an integer between 1 and 2147483646, since it is used by the pseudo-random number generator.

The version of USERSI in Files 1 and 2 is specialized to photon correlation spectroscopy and is discussed in Sec 6. However, the comment cards explain how to simulate data following normal and Poisson statistics, and how to use RUSER(3) to set the noise level.

`USEREX`	(only called when `SIMULA=T`) computes the noise-free value of a simulated data point.

As Fortran arguments in `USEREX` you are supplied with `NYMAX`, `T` (the array of the t_{kn}), `NY` (the array of the N_{y_n}), and with `K` $= k$ and `N` $= n$ (the subscripts of the t_{kn}), and in case of a convolution `RESPON` may contain the response function. `NYMAX` is only used in a `DIMENSION` statement for `T`. You must set `USEREX` to the noise-free value of the data point at `T(K,N)`.

In the version of `USEREX` in Files 1 and 2, the noise-free data is a simple sum of exponentials plus background,

$$\texttt{USEREX = RUSER(30) + } \sum_{\texttt{J=1}}^{\texttt{IUSER(10)}} \texttt{RUSER(19+J)} \cdot \exp\{\texttt{-RUSER(9+J)} \cdot \texttt{T(K,N)}\}$$

where you must have specified the number of exponentials, the amplitudes, the decay constants, and the background appropriately.

4.1.7.3 Criteria for convergence

The following termination criterion for stopping the iterative search for the minimum of `VAR` (Eq (4.1.2–2)) has proved to work well:

$$|\texttt{VAR} - \texttt{VAR}_{\text{old}}| \le \max\{\texttt{VARMIN}, \texttt{VAR}_{\text{old}} \cdot \texttt{CONVRG}\} \qquad (1)$$

where $\texttt{VAR}_{\text{old}}$ is the variance from the last iteration and

$$\texttt{VARMIN} = \sum_{n=1}^{N_D} \sum_{k=1}^{N_{y_n}} (w_{kn} y_{kn} \cdot \texttt{CONVRG})^2$$

and `CONVRG` is described below.

`CONVRG`	defines the maximum relative accuracy of your data. The default value `CONVRG=5·10`$^{-5}$ is assigned in the `BLOCK DATA` subprogram, i.e., your data has at most 4 figures accuracy.

The criterion of Eq (1) says, that the iterations are stopped as soon as no significant progress is made in reducing the value of `VAR`. SPLMOD, however, continues unless no significant progress has been made over `MCONV` iterations.

`MCONV`	specifies how often the termination criterion of Eq (1) has to be satisfied before the iterative search will be stopped.

Sometimes at the end of the analysis when the variance is already close to the minimum, or when the normal equations matrix is nearly singular, it can happen that the variance `VAR` is increased. To prevent the program from iterating endlessly SPLMOD allows only `NABORT` times the variance being successively increased.

`NABORT`	specifies how often the variance `VAR` is allowed to increase successively, before the iterative search is stopped.

Usually it is not necessary to change one of the three Control Variables `CONVRG`, `MCONV`, or `NABORT`.

`MXITER` (2)	`MXITER(ISTAGE)` is an upper bound for the number of iterations allowed at Stage `ISTAGE`, where Stage 1 is the `PRELIMINARY UNWEIGHTED ANALYSIS` and Stage 2 is the `FINAL ANALYSIS`.

The values `MXITER(1)=20` and `MXITER(2)=40` are recommended.

4.1.7.4 Choosing starting values

Because of the reduction in dimensionality when using spline approximations (`DOSPL=T`), as already described in Sec 3.2, iterations are cheap. Therefore SPLMOD performs a thorough and reliable systematic search for the minmum of `VAR` (Eq (4.1.2–2)), trying different starting values for the λ_j, $j = 1, \ldots, N_\lambda$. Naturally you have a better chance of finding the global minimum rather than a local one when trying a lot of different starting values.

`MTRY` (10,2)	`FORMAT(20I4)` `MTRY(J,ISTAGE)` defines the maximum number of tries that will be made to find a solution for a single N_λ = `NL(J)` at Stage `ISTAGE`, where Stage 1 is the `PRELIMINARY UNWEIGHTED ANALYSIS` and Stage 2 is the `FINAL ANALYSIS`. `MTRY` must satisfy Eq (3.5–8).

The strategy for choosing the starting values is as follows: the interval of allowed values for the λ_j, defined by the constraints `PNMNMX`, is divided into M_{box}

equally spaced subintervals and M_{box} is chosen satisfying the inequality relation $\binom{M_{\text{box}}}{N_\lambda} \leq$ MTRY. Then all the possible combinations of taking the value at the center of N_λ subintervals out of M_{box} as starting values, are performed.

While iterating SPLMOD keeps track of the path through the $\{\lambda_j\}$ parameter space, thus avoiding starting values which were already tried during other iterations.

When using `DOSPL=.TRUE.` (the default value) `MTRY(1,1)=10`, `MTRY(1,2)=20` and `MTRY(J,1)=20`, `MTRY(J,2)=50` for `NL(J)=J` and `J=1,...,NNL` are recommended.

The same strategy is used with `DOSPL=.FALSE.`, i.e., using the exact model. However, the computer time is often almost directly proportional to **MTRY**, thus if the size of the problem makes computer time a major consideration, `MTRY(J,I)=5`, for `J=1,...,NNL` and `I=1,2` should be adequate.

SPLMOD also allows you to input your own starting values for a single N_λ and at Stage 1 or 2, not regarding the value of MTRY. This is initiated by the Control Variable DOSTRT and you have to set the starting values in the USER subprogram USERST described below.

DOSTRT (10,2)	FORMAT(20L1)	DOSTRT(J,ISTAGE)
= T	if USERST has to be called to input your starting values for a single N_λ = NL(J), at Stage ISTAGE.	
= F	if USERST has not to be called.	

USERST	(only called when DOSTRT=T) gives you the possibility to input your own starting values for the nonlinear parameters λ_j.

As Fortran arguments in USERST, you are supplied with ISTAGE, NYMAX, T (the array of the t_{kn}), NY (the array of the N_{y_n}), NONLMX, and PLAM (the array of the starting values for the λ_j). NYMAX and NONLMX are used in a DIMENSION statement.

The version of USERST in Files 1 and 2 illustrates the case that you have stored the starting values in USER array RUSER, i.e., $\lambda_j = r_{50+j}$ for $j = 1, \ldots, N_\lambda$, and r_{50+j} = RUSER(50+J). Note, that you put $H(\lambda_j)$ (Sec 4.1.3.1) in the array PLAM and *not* λ_j.

4.1.7.5 Remark on minimization METHOD

As already mentioned in Section 4.1.2, we try to find a least value of the variance VAR (Eq (4.1.2–2)). One way of locating a stationary point is to use *Newton*'s classical minimization method. The main disadvantage with this method is, that it needs second derivatives of the residuals, which may be expensive to calculate. This can be avoided by simply omitting the terms with second derivatives. The latter is usually referred to as *Gauß-Newton* method.

Although SPLMOD is able to perform the classical *Newton* method, it will normally use a modified *Gauß-Newton* method, which takes advantage of the fact, that the parameters are separable.

METHOD	= 2	use modified *Gauß-Newton*
	= 4	use classical *Newton*

The value of **METHOD=2** is strongly recommended.

4.2 More on the Composition of a Data Set

Some of the USER subprograms may require input data. Below is listed the order in which any input data must occur in the input data deck.

Card 1 – Card Set 7 (standard input data - see Sec 3.3.2)	
Card Set 8	USERIN
Card Set 9	USERSI
Card Set 10	USEREX
Card Set 11a	USERST
Card Set 12	USERWT
Card Set 11b	USERST (only if IWT=2, 3, or 5)

Obviously you only need a Card Set if the corresponding USER subprogram is called and if it requires input data.

If IWT=2, 3, or 5 and you require Card Set 11a, then you can save inputting the same data again in Card Set 11b by storing the data as Card Set 11a is read and using a flag to remember this and to skip the input of Card Set 11b. You can also put the necessary data into a USER array with Card Set 2 or in USERIN and thereby avoid using Card Set 11a completely.

4.3 Calling Diagram for the Subprograms

It is not necessary to read this section to use SPLMOD. With the diagram below, you can trace the calling sequences for all the subprograms for the version of SPLMOD in Files 1 and 2. In Files 1 and 2, the comment cards at the beginning of each subprogram usually give you at least a rough idea of the tasks being performed there.

In the diagram below, **MAIN** and **BLOCKD** represent the main subprogram and the BLOCK DATA subprogram, respectively. All the other names in rectangles are the actual names of subprograms, and each name only appears once. Connecting lines run to the right from the rectangle of a subprogram indicating all the subprograms that it calls. The calling sequence is approximately from top to bottom.

Each oval is a block of one or more subprograms that is called in more than one place in the diagram. Each oval is defined by a separate diagram to the right or below where the oval is called. In the diagram of an oval, the number in parentheses next to the oval is the number of places the oval appears in the other diagrams above and to the left.

- MAIN
  - BLOCKD
  - INIT
    - DIFF
    - ERR
  - INPUT
    - STORIN
      - ERR
    - ERR
    - READYT
      - ERR
      - USERIN
    - WRITIN
      - WRITYT
  - USERSI
    - USEREX
    - RGAUSS
      - RANDOM
  - WRITYT
  - ERR
  - USERTR
  - CHOTRD
  - INICAP
    - BCOEFF
      - USERFN
      - CHOSOL
    - USERFL
  - ANALYZ
    - NPASCL
    - USERST
      - ERR
      - USERTR
    - USEROU
    - NOKSUB
    - LSTSQR
      - VARIAN
      - USEROU
      - OUTITR
        - USERTR
      - ERR
      - HESEXA
      - HESSPL
      - PIVOT
      - INTERP
        - ERR
        - VARIAN
    - ERR
    - OUTALL
      - ERR
      - FULLSM
        - HESEXA
        - HESSPL
      - PIVOT
      - OUTPAR
        - USERTR
      - USEROU
      - OUTCOR
      - VARIAN
      - GETPRU
        - PGAUSS
      - PLRES
        - ERR
      - PLPRIN
    - VARIAN
    - FISHNI
      - ERR
      - BETAIN
        - ERR
        - GAMLN
    - USERTR
  - SETWT
    - USERWT
    - ERR

- VARIAN (4)
  - VARIAN
    - LINPAR
      - INTERV
      - BSPLIN
      - LINEQS
        - USERFN
        - USERFL
      - PIVOT
    - USERFN
    - USERFL
    - ERR
    - NPASCL

- PIVOT (3)
  - PIVOT
    - PIVOT1

- HESEXA (2)
  - HESEXA
    - USERFN
    - USERFL
    - HESSEN

- HESSPL (2)
  - HESSPL
    - BSPLIN
    - HESSEN

- USERFL (4)
  - USERFL
    - ERR

- USERFN (4)
  - USERFN
    - ERR
    - USERTR

- USEROU (3)
  - USEROU
    - ERR
    - USERTR

- USERTR (7)
  - USERTR
    - ERR

- BSPLIN (2)
  - BSPLIN

- ERR (19)
  - ERRMES

- HESSEN (2)
  - HESSEN

- NPASCL (2)
  - NPASCL

- WRITYT (2)
  - WRITYT

5. DIAGNOSTICS AND HINTS FOR TROUBLE - SHOOTING

5.1 Nonstandard Diagnostics and Aborts

SPLMOD performs many tests for error conditions, and there are more than 40 diagnostics. Most of these take a standard form, "`ERROR` Name n. `(CHECK USERS GUIDE.)`", where Name is the name of the subprogram where the error occurred and n is an integer. If one of these messages occurs, see Sec 5.3, where they are explained.

However, certain errors, especially during input, can occur before there is a chance for a diagnostic. For example if `IFORMY`, `IFORMT`, or `IFORMW` do not contain a legal Fortran specification enclosed in parentheses, then the run may abort with no error message.

If an end-of-file is encountered while trying to read input data, then check the following:

- You must have an `END` card (Card 3 in Sec 3.3.2).
- Has a Control Variable been set so that a `USER` subprogram requiring input data has been inadvertently called?
- The input and output files `NIN` and `NOUT` must be properly opened (Sec 2.3.4) if your system does not do this automatically for you. Similarly, if you are using a scratch file, then file `IUNIT` must be properly opened (Sec 4.1.7.1).

As discussed in Sec 2.3.2, you must insure that your computer will replace underflows with zero; many underflows are expected to occur.

Much effort has been made to avoid overflows or division by zero. Overflows are usually caused by improper scaling.

There are a few diagnostics that do not take the standard form. Several of these are in the `USER` subprograms, but they are self-explanatory. You should include ample diagnostics in your versions of `USER` subprograms. SPLMOD checks each card in Card Set 2 to make sure that it does not attempt to change Name n if n exceeds the `DIMENSION` specification of the Control Variable Name.

5.2 General Hints for Trouble - Shooting

Usually the cause of the diagnostic is apparent. Often it is due to incorrect input data or to a forbidden combination of Control Variables or `USER` subprograms. Sometimes, however, the cause is not obvious. There may not even be a diagnostic or an abort, just meaningless results. One possiblity is that there is an error in SPLMOD. It is obviously impossible to test all combinations of Control Variables and `USER` subprograms. You should send us your output, as well as compilation listings of any subprograms that you have modified, information on the version used, date, and excessive amounts of data in machine-readable form. This will help us maintain SPLMOD.

First, however, you should make sure that **you** have not caused the error. Some suggestions:

(1) Check all changes that you have made. To be sure, you might want to make the changes again to a copy of your original file copy of SPLMOD and rerun.

(2) In your versions of the `USER` subprograms, have you exceeded any of the `DIMENSION` specifications of any arrays? The actual `DIMENSION` specifications are given in the `MAIN` subprogram (see also Sec 3.5).

(3) In your versions of the `USER` subprograms, have you changed any of the `COMMON` variables to illegal values? It is not recommended that you change these ever, except in the few cases where you are explicitly instructed to do so in a `USER` subprogram. Extensive tests on these variables are made right after the input, but this does not help if you change them to illegal values later.

(4) Do the actual `DIMENSION` specifications in Lines `MAIN0064`–`MAIN0081` of the main subprogram correspond to those in the `DATA` statement in Lines `MAIN0140`–`MAIN0147` as specified in Lines `MAIN0085`–`MAIN0113`. This is of course crucial, since the `DIMENSION` specifications passed to the subprograms would otherwise be wrong.

(5) Grossly erroneous y_k, t_k, constraints, etc. can cause problems, sometimes without diagnostics. For example, if all your y_k were negative, but your constraints implied that they could only be positive, then all the analyses might fail to converge.

(6) A run with very simple simulated data (e.g., with a high signal-to-noise ratio) could be helpful.

(7) A run with a good debugging compiler like `WATFIV` (Sec 2.3.3) could be helpful.

5.3 Standard Diagnostics

The standard diagnostics are of the form:

`ERROR` Name n. `(CHECK USERS GUIDE.) ********`,

where Name is the name of the subprogram where the error occurred and n is an integer that uniquely identifies the error in Name. These diagnostics are listed below (with Name in alphabetical order). Following Name n is one of the following three abbreviations in parentheses:

`(F)` means a fatal error. The run is aborted.

`(NF)` means a non-fatal error. The run continues, possibly after taking corrective action. Usually the corrective action is to skip part of the analysis immediately following the error and jump to a new part.

`(W)` means a warning. SPLMOD has detected a peculiar condition, but it is not necessarily an error. The run continues.

The error or warning is then explained, and *possible remedies* are given after the abbreviation **PR**: . Several possible remedies numbered (1), (2), etc. are alternatives in descending order of priority; generally only one is necessary to correct the problem, not all of them.

If the PR do not help, then try the suggestions in Sec 5.2. If these suggestions do not help, then send me the information listed in Sec 5.2. Also send me this information if the error is described as an *Illogical Stop*. This indicates a condition that is apparently due to a programming error of mine. Of course, if you make certain wide-ranging errors in your modifications of SPLMOD, like those described in points (2), (3), and (4) in Sec 5.2, then nearly anything can happen, perhaps including an Illogical Stop. So you should first check the points in Sec 5.2 when the PR does not help or an Illogical Stop occurs.

Cross references are usually not made to the equations in Secs 3.1 and 3.2 when symbols defined there are used or to other sections when terms indexed in Sec 8 are used. So you should use these indexes.

`MAIN 1` `(F)` One of the following requirements is not met:

`PLMNMX(1) < PLMNMX(2)`

`PNMNMX(1) < PNMNMX(2)`

where `PLMNMX` and `PNMNMX` are the actual constraints for the linear parameters α_j, γ_μ, and the nonlinear parameters λ_j, respectively, possibly modified in `USER` subprogram `USERIN`. These constraints are output immediately after the three machine-dependent parameters `PRECIS`, `SRANGE`, and `RANGE` (Sec 3.6.3).

PR: (1) Input appropriate values for Control Variables `PLMNMX` and `PNMNMX`.

(2) Correct `USER` Subprogram `USERIN`.

`ANALYZ 1` `(NF)` Variance `VAR` (Sec 3.2) increased the last `NABORT` times. This can occur if the normal equations matrix is nearly singular, or sometimes at the end of the analysis when the variance is already very close to the minimum.

PR: (1) Is `NABORT` set to a proper value (e.g., `NABORT=3`)?

(2) Is `PNMNMX` chosen properly, and according to it the starting values for λ?

`ANALYZ 2` `(NF)` Maximum iteration without convergence.

PR: (1) Check `MXITER`.

(2) If `VAR` (Sec 3.2) is decreasing monotonically, then you can try a greater value for `MXITER`, otherwise it is hopeless.

`ANALYZ 3` `(NF)` All diagonal elements of the nonlinear least squares matrix are unpivoted, (i.e., singular normal equations matrix).

PR: (1) Check `PNMNMX`, if the allowed region for the λ is too wide, SPLMOD will potentially have tried unreasonable starting values, if the allowed region is chosen too restrictive or even impossible for the data at hand, then the diagonal elements are unpivoted because of boundary violations.

(2) Scaling.

`ANALYZ 4` `(W)` Some of the correction steps for the parameters are set to zero because of (1) boundary conditions or (2) a nearly singular normal equations matrix.

`ANALYZ 5` `(NF)` `VAR=RANGE` (Sec 3.2 and 2.3.1) in all `MTRY` tries for a single value of N_λ.

PR: (1) Check data for gross errors.

(2) Decrease N_λ.

ANALYZ 6 (NF) All analyses failed to converge (VARBES=RANGE (Sec 3.2 and 2.3.1), i.e., the best variance found so far is still greater than RANGE). If this occurs in the preliminary analysis, SPLMOD will set IWT=1 and try a final analysis.

PR: (1) Check data for gross errors.

(2) Decrease N_λ.

BETAIN 1 (F) The arguments of **BETAIN** are out of range. This should only occur if **NY** ≥ 40000; otherwise it is an Illogical Stop.

PR: Reduce the maximum of **NY** to less than 40000.

BETAIN 2 (F) Illogical Stop. It can only occur if convergence to the incomplete beta function was still not attained after 20000 iterations.

FISHNI 1 (W) One of the degree-of-freedom arguments in Eq (3.6.5–1) is not positive. This can happen if the number of degrees of freedom equals the number of data points. In that case the **PROBABILITIES THAT OTHER SOLUTIONS ARE WORSE:** have no meaning and the value **1.0** is arbitrarily assigned to it. The run continues.

PR: Reduce N_λ or N_γ so that
$(N_\lambda + N_\gamma) * N_D + N_\lambda < \max\{\mathtt{NY}(n)\}$ for $n = 1, \ldots, N_D$.

INIT 1 (F) The accuracy of the **REAL** arithmetic is less than about 5 significant figures (if you are using Version 3SP) or the accuracy of **DOUBLE PRECISION** is less than about 9 figures (if you are using Version 3DP).

PR: In the likely case that your computer really has more accuracy, this is an Illogical Stop. Otherwise, try changing Versions.

INPUT 1 (F) The card (in your Card Set 2) printed above does not have a valid Control Variable name in columns 2–7. This is a fatal error, but SPLMOD will continue checking the rest of the data set for errors until it has found **MIOERR** errors or until it has reached the end of the data set.

(This is the most common error. See Secs 3.3 and 3.4.)

PR: (1) Correct the name of the Control Variable.

(2) Did you erroneously use two cards for the preceding Control Variable when it requires one only?

(3) Check that Cards 1 and 3 are correct.

INPUT 2 (F) At least one of the following requirements is not met:

$1 \leq \mathtt{ND} \leq \mathtt{NDMAX}$

$1 \leq \mathtt{NNL} \leq 10$

$-2 \leq \mathtt{IRWCAP} \leq 2$

$1 \leq \mathtt{IWT} \leq 5$

$0 \leq \mathtt{NGAM} \leq \mathtt{NGAMMX}$

$18 \leq \mathtt{NZ} \leq \mathtt{NZMAX} - 2$

$0 \leq |\mathtt{IPLFIT(J)}|, \mathtt{IPLRES(J)}, \mathtt{IPRITR(J)} \leq 3\ , \quad \mathtt{J} = 1, 2$

$1 \leq \mathtt{IPRINT(I,J)} \leq 3\ , \quad \mathtt{I} = 1, \ldots, 3, \quad \mathtt{J} = 1, 2$

$1 \leq \mathtt{MTRY(I,J)} \leq \mathtt{MTRYMX}\ , \quad \mathtt{I} = 1, \ldots, \mathtt{NNL}, \quad \mathtt{J} = 1, 2$

In the following call $\mathtt{MAXNL} := \max\{\mathtt{NL(J)}\}$, $\mathtt{J} = 1, \ldots, \mathtt{NNL}$

$0 < \mathtt{MAXNL} < \mathtt{NONLMX}$

$0 < \mathtt{MAXNL} + \mathtt{NGAM} < \mathtt{NLINMX}$

$(\mathtt{MAXNL} + \mathtt{NGAM}) * \mathtt{ND} + \mathtt{MAXNL} < \mathtt{NLNDMX}$

$(\mathtt{MAXNL} + \mathtt{NGAM}) * \mathtt{ND} + \mathtt{MAXNL} < \mathtt{NYSUM}$

$\max\{\mathtt{MCONV}, \mathtt{NABORT}\} \leq 50$

You can directly check all of these relations because the final values of the Control Variables will have been output (as on *Page 3* of the test output).

INTERP 1 (NF) *Gauß*-vector of a correction step points in the wrong direction.

LSTSQR 1 (NF) Same as **ANALYZ 1**.

LSTSQR 2 (NF) Same as **ANALYZ 2**.

LSTSQR 3 (NF) Same as **ANALYZ 3**.

OUTALL 1 (NF) Singularity in inverting full normal equations matrix. No correlations of parameters are calculated.

OUTALL 2 (NF) All diagonal elements of full normal equations matrix are unpivoted during inversion. No correlations of parameters are calculated, no residuals are plotted, and no data and fit to data.

PR: See PR in **ANALYZ 3**.

OUTALL 3 (NF) Negative diagonal element in inverted full normal equations matrix detected, i.e., the matrix is not positive definite.

PR: (1) Use **METHOD=2**.

(2) Restrict the interval of the allowed values for the λ, i.e., change **PNMNMX**.

PLRES 1 (NF) The plot of the residuals is being skipped because **LINEPG** < 17 and this does not provide enough space per page for the plot.

PR: (1) Increase **LINEPG**.

(2) Do without the plot by setting **IPLRES**(J)=0, J=1,2.

PLRES 2 (W) The plot of the residuals is being interrupted after **MPAGE** (=30) pages of output from this single plot.

PR: (1) Do without the plot by setting **IPLRES**(J)=0, J=1,2.

(2) If $\max\{\mathtt{NY}(n)\}$ for $n = 1, \ldots, N_D$ is so large and you still want plots of the residuals, then increase **MPAGE** in the **DATA** statement in Line **PLRES013**.

READYT 1 (F) Card 4a (Sec 3.3.2) does not contain the characters **NINTT** followed by 1 blank in columns 2–7 or Card 5a (Sec 3.3.2) does not contain the characters **NY** followed by 4 blanks in columns 2–7. The card is printed above the diagnostic.

PR: (1) Correct columns 2–7.

(2) Are Card 3 and Card 4a or Card 5a in the proper positions in the input data deck?

READYT 2 (F) While reading Card 4a or Card 5a (Sec 3.3.2), a card was encountered with **NINTT** ≤ 0 or **NY** ≤ 0 respectively. The card is printed above the diagnostic.

PR: (1) Is **NINTT** correct?

(2) Is **NY** correct?

(3) Are Card 3 and Card 4a or Card 5a in the proper positions in the input data deck?

READYT 3 (F) While reading Card Set 4b (Sec 3.3.2), a card without the characters **NSTEND** in columns 2–7 was encountered. The card is printed above the diagnostic.

PR: (1) Correct columns 2–7.

(2) Is **NINTT** correct?

(3) Are Card 3 and Card Set 4b in the proper positions in the input data deck?

READYT 4 (F) While reading Card Set 4b (Sec 3.3.2), a card was encountered with **NT** < 2 or **NT** so large that **NY** would exceed **NYMAX** and therefore violate Eq (3.5–6). The card is printed above the diagnostic.

PR: (1) Correct **NT**. (You can always use **NT** at least 2, regardless of the spacing of the **T** values.)

(2) If the spacing of the **T** values is very irregular, it may be easier to use Card 5a and Card Set 5b.

(3) If **NT** is correct, then increase **NYMAX** (Sec 3.5).

READYT 5 (F) **NY** on Card 5a (Sec 3.3.2) exceeds **NYMAX** and therefore violates Eq (3.5–6). Card 5a is printed above the diagnostic.

PR: (1) Correct **NY**.

(2) Increase **NYMAX** (Sec 3.5).

READYT 6 (F) You have input a negative least squares weight. The **W(K,N), K=1,NY(N)** for that **N** with which the error occurred, are printed after the diagnostic.

PR: (1) If you used **DOUSIN=T** and computed the weights in **USERIN**, then correct this.

(2) Otherwise, you must have used **IWT=4**; correct **IFORMW** or Card Set 7 (Sec 3.3.2).

SETWT 1 (F) Illogical Stop. This can only occur if **IWT** is not **2**, **3**, or **5** in **SETWT**.

`SETWT 2 (F)` `IWT=2` or `3` and $(\mathtt{YSAFE})_{kn} = \mathtt{0.0}$ (see Sec 4.1.3.4). Therefore the expression for w_{kn} would be `1.0/0.0` and division by zero would be attempted. This is very unlikely, since all `NERFIT` residuals as well as $(\mathtt{YFIT})_{kn}$ would have to be zero.

PR: Increase `NERFIT` or change `IWT`.

`STORIN 1 (NF)` For `LOGICAL` Control Variables, the only allowed values are `1.0` (for `.TRUE.`) and `-1.0` (for `.FALSE.`).

PR: Correct your input data.

`STORIN 2 (NF)` The subscript of one of the dimensioned Control Variables (e.g., `RUSER`, `IUSER`, `LUSER`, see Sec 8.2 for more) is out of range.

PR: (1) Correct your input data.

(2) Did you modify the `DIMENSION` of one of the `RUSER`, `IUSER`, or `LUSER` Control Variables?

`VARIAN 1 (NF)` `VAR` (Sec 3.2) is negative due to rounding errors. This can only occur when using `DOSPL=.TRUE.` (Sec 4.1.6) and especially when using `SINGLE PRECISION` version.

PR: (1) Use `DOUBLE PRECISION` version.

(2) Increase `NZ`, the number of interpolation knots.

(3) Restrict the interval of the allowed values for the λ, i.e., change `PNMNMX`.

(4) Use `DOSPL=.FALSE.` .

`USERFL 1 (F)` Illogical Stop. This can only occur if `N` > `ND`, where `N` is the 2nd subscript of the data point in the program calling `USERFL`.

`USERFL 2 (F)` Illogical Stop. This can only occur if `K` < 0 or `K` > `NY(N)`, where `K` is the 1st subscript of the data point in the program calling `USERFL`.

`USERFL 3 (F)` This is specialized to the versions of `USERFL` in Files 1 and 2. The requirement 1 ≤ `MU` ≤ 6 is violated, where `MU` is the subscript μ in Eq (3.1–1). If you modify `USERFL`, you should modify the requirement to be 1 ≤ `MU` ≤ `NGAM`.

PR: Correct `USERFL` or `NGAM`.

`USERFL 4 (F)` This is specialized to the versions of `USERFL` in Files 1 and 2. Only certain combinations of `NGAM`, `IUSER(1)`, and `LUSER(1)` are allowed, for further details see Sec 6.1.

PR: Input correct Control Variables.

`USERFL 5 (F)` This is specialized to the versions of `USERFL` in Files 1 and 2. Only certain combinations of `NGAM`, `IUSER(1)`, and `LUSER(1)` are allowed, for further details see Sec 6.1.

PR: Input correct Control Variables.

`USERFL 6 (F)` This is specialized to the versions of `USERFL` in Files 1 and 2. Only certain combinations of `NGAM`, `IUSER(1)`, and `LUSER(1)` are allowed, for further details see Sec 6.1.

PR: Input correct Control Variables.

`USERFN 1 (F)` This is specialized to the versions of `USERFN` in Files 1 and 2, and is only of interest when using recursion formulas. The requirement 1 ≤ `J` ≤ `NZ` + 2 is violated.

PR: Correct `USERFN`.

`USERFN 2 (F)` Illogical Stop. This can only occur if `N` > `ND`, where `N` is the 2nd subscript of the data point in the program calling `USERFN`.

`USERFN 3 (F)` Illogical Stop. This can only occur if `K` < 0 or `K` > `NY(N)`, where `K` is the 1st subscript of the data point in the program calling `USERFN`.

`USERFN 4 (F)` Illogical Stop. This can only occur if `IDERIV` < 0 or `IDERIV` > 2, where `IDERIV` says which derivative of `USERFN` is to compute.

PR: Correct `USERFN`.

`USERFN 5 (F)` This is specialized to the versions of `USERFN` in Files 1 and 2, the requirement 1 ≤ `IUSER(1)` ≤ 3 is violated, where `IUSER(1)` indicates if $f(\lambda, t)$ in Eq (3.1–1) is a pure exponential (`IUSER(1)=1`) or a convolution (`IUSER(1)=2` or `3`). For further details see Sec 6.1.

PR: Correct `USERFN`.

`USERFN 6 (F)` This is specialized to the versions of `USERFN` in Files 1 and 2. For `IUSER(1)=3` is no 2^{nd} derivative available in `USERFN` and therefore Control Variable `METHOD` has to be less than 4 (See Sec 4.1.7.5 for an explanation of `METHOD`). For further details see Sec 6.1.

PR: (1) Use `METHOD=2`.

(2) Change `USERFN`.

`USEROU 1 (F)` Illogical Stop. This can only occur if `ISTAGE` is not `1` or `2`.

PR: Correct `USEROU`.

`USEROU 2 (F)` This is specialized to the versions of `USEROU` in Files 1 and 2. The requirement $1 \leq$ `IROUTE` ≤ 6 is violated, where `IROUTE` specifies what has to be output.

PR: Correct `USEROU`.

`USERST 1 (F)` Illogical Stop. This can only occur if `ISTAGE` is not `1` or `2`.

PR: Correct `USERST`.

`USERST 2 (NF)` This is specialized to the versions of `USERST` in Files 1 and 2 and can only occur if your starting values don't lie within the boundaries `PNMNMX(1)` and `PNMNMX(2)`.

PR: Check your own starting values.

`USERTR 1 (F)` Illogical Stop. This can only occur if the actual argument `IFUNCT` in the program calling `USERTR` is not `1`, `2`, or `3`.

`USERTR 2 (F)` Illogical Stop. This can only occur if `USERTR` is called with an improper argument `X`, which causes overflow or other errors aborting the program.

6. APPLICATIONS

6.1 Examples for USERFN and USERFL

One of the main applications of SPLMOD is the analysis of fluorescence lifetime measurements where the data is represented by a sum of exponentials. Below an example of **USER** subprogram **USERFN** is given for this case and the parameter grid for the nonlinear parameters λ_j is assumed to be in equal logarithmic intervals,

```
USERFN=0.
EZ=USERTR(Z,2)*T(K,N)
IF (EZ .GE. EXMAX) RETURN
USERFN=EXP(-EZ)
IF (IDERIV .EQ. 0) RETURN
USERFN=-EZ*USERFN
IF (IDERIV .EQ. 1) RETURN
USERFN=USERFN*(1.-EZ)
RETURN
```

and **EXMAX**=ln(**SRANGE**) is the largest reasonable exponent in **EXP** which does not cause underflow or overflow.

In nanosecond fluorescence decay the data must be represented by a convolution of the instrumental response function, say $E(t)$, with the actual fluorescence, say $F(t)$:

$$y(t) = E(t) * F(t).$$

The calculation of the above convolution can be quite expensive. Therefore considerable reduction in computational time may be achieved by use of recursion formulas [16], thereby replacing the convolutional integral by a discrete sum, applying trapezoidal rule.

Though it is possible to use recursion relations for nonregular spaced t_k, it is mostly efficient for equally spaced t_k or if the t_k can be input in groups of equally spaced t_k.

Another difficulty may arise with convolutions. Because of small but important drifts in the instrument with time and the dependence of the photomultiplier on count rate and wavelength, it is very difficult to measure $E(t)$ directly. To circumvent this the fluorescence decay of a standard system that is known to have a monoexponential decay curve, $y_r(t)$, is measured, which is called *reference* function in the sequel. *Gauduchon* & *Wahl* [15] used this technique

for multiexponential decay. Both $y_r(t)$ and $y(t)$ have the same $E(t)$ because they are measured simultaneously with the same detector [6].

The same kind of formulas are obtained as Eq (3.1–1),

$$y(t) = \gamma_r y_r(t) + \sum_{j=1}^{N_\lambda} \beta_j y_r(t) * \exp(-\lambda_j t) + \sum_{\mu=1}^{N_\gamma} \gamma_\mu g_\mu(t) \tag{1}$$

except for modified amplitudes,

$$\beta_j = \alpha_j(\lambda_r - \lambda_j)/\alpha_r, \tag{2}$$

and additionally the extra γ_r,

$$\gamma_r = \sum_{j=1}^{N_\lambda} \alpha_j/\alpha_r \tag{3}$$

where α_j are the original amplitudes and α_r and λ_r are the amplitude and the lifetime of the standard respectively. Normally λ_r is known and α_r is absorbed into the α_j. The analysis is then performed with λ_r and γ_r fixed, i.e., with $2N_\lambda + N_\gamma$ free parameters γ_μ, β_j, and λ_j. However Eq (1) can also be analyzed for the $2N_\lambda + N_\gamma + 1$ parameters γ_μ, β_j, λ_j, and γ_r. Eq (2) and (3) can then be solved for λ_r; e.g., when $N_\lambda = 1$, we have $\lambda_r = \lambda_1 + \beta_1/\gamma_r$. This can be useful to test the reliability of the instruments and standards, by using a second standard with known λ_1 as the sample and seeing if the correct lifetimes come out of the analysis. Otherwise, fixing λ_r and γ_r is recommended. The sum over the linear functions $g_\mu(t)$ in Eq (1) contains two terms, a constant to account for background in the data and

$$\gamma_s[\lambda_r y_r(t) + y_r'(t)]$$

to account for small errors in determining the time delay between the sample and reference curves (prime denotes differentiation). For a detailed discussion of the above equations see [6].

All above described possibilities are available in the versions of **USERFN** and **USERFL** in Files 1 and 2. The following diagram shows how to set Control Variables **NGAM**, **IUSER(1)**, **RUSER(1)**, and **LUSER(1)** to use one of the above mentioned approaches.

Case	NGAM	IUSER(1)	RUSER(1)	LUSER(1)
1	0	1	0.	F
2	1	1	0.	F
3	0	2	0.	F
4	1	2	0.	F
5	2	2	0.	F
6	1	3	λ_r	F
7	2	3	λ_r	T

The meaning of the different cases is described as follows:

Case (1) pure exponentials, no background.

Case (2) pure exponentials plus constant background.

Case (3) convolution, no background; use response function to deconvolute.

Case (4) convolution with constant background; use response function to deconvolute.

Case (5) convolution with constant background; use reference function to deconvolute, λ_r *not* known, correct *not* for shift between sample and reference.

Case (6) convolution with constant background; use reference function to deconvolute, λ_r known, correct *not* for shift between sample and reference.

Case (7) convolution with constant background; use reference function to deconvolute, λ_r known, and correct for shift between sample and reference. The error ϵ in the shift can then be estimated from the least squares parameters by

$$\epsilon = \gamma_2 \,/\, \sum_{j=1}^{N_\lambda} \alpha_j$$

if $\epsilon\,\lambda_j \ll 1$ for all observable λ_j.

7. REFERENCES

[1] Provencher, S.W. and Vogel, R.H., Regularization techniques for inverse problems in molecular biology, in *Progress in Scientific Computing*, Deuflhard, P. and Hairer, E., eds. (Birkhäuser, Boston, 1983) Vol. 2, 304–319.

[2] Provencher, S.W., Inverse problems in polymer characterization: Direct analysis of polydispersity with photon correlation spectroscopy, *Makromol. Chem.* **180**: 201–209 (1979).

[3] Provencher, S.W., A constrained regularization method for inverting data represented by linear algebraic or integral equations, *Comput. Phys. Commun.* **27**: 213-227 (1982).

[4] Provencher, S.W., CONTIN: A general purpose constrained regularization program for inverting noisy linear algebraic and integral equations, *Comput. Phys. Commun.* **27**: 229–242 (1982).

[5] Provencher, S.W., DISCRETE: A FORTRAN IV program for the automatic analysis of multicomponent exponential decay data, *Technical Report*, Max-Planck-Institut für biophysikalische Chemie, D-3400 Göttingen-Nikolausberg (1976).

[6] Wijnaendts van Resandt, R.W., Vogel, R.H., and Provencher, S.W., Double beam fluorescence lifetime spectrometer with subnanosecond resolution: Application to aqueous tryptophan, *Rev. Sci. Instrum.* **53**: 1392–1397 (1982).

[7] Provencher, S.W. and Vogel, R.H., Information loss with transform methods in system identification: A new set of transforms with high information content, *Math. Biosci.* **50**: 251–262 (1980).

[8] Provencher, S.W., An eigenfunction expansion method for the analysis of exponential decay curves, *J. Chem. Phys.* **64**: 2772–2777 (1976).

[9] Provencher, S.W., A Fourier method for the analysis of exponential decay curves, *Biophys. J.* **16**: 27–41 (1976).

[10] Abramowitz, M. and Stegun, I.A., editors, *Handbook of Mathematical Functions*, (Dover Publications, New York, 1972).

[11] Nave, C., Brown, R.S., Fowler, A.G., Ladner, J.E., Marvin, D.A., Provencher, S.W., Tsugita, A., Armstrong, J., and Perham, R.N., Pf1 filamentous bacterial virus: X-ray fibre diffraction analysis of two heavy atom derivatives, *J. Mol. Biol.* **149**: 675–707 (1981).

[12] Draper, N.R. and Smith, H., *Applied Regression Analysis*, (Wiley, New York, 1966).

[13] de Boor, C., *A Practical Guide to Splines*, (Springer, New York, 1978) Chap. 11 and 16.

[14] Price, P.F., A comparison of least-squares and maximum-likelihood estimators for counts of radiation quanta which follow a Poisson distribution, *Acta Cryst. A* **35**: 57–60 (1979).

[15] Gauduchon, P. and Wahl, P.H., Pulsefluorimetry of tyrosyl peptides, *Biophys. Chem.* **8**: 87–104 (1978).

[16] Grinvald, A. and Steinberg, I.Z., On the analysis of fluorescence decay kinetics by the method of least squares, *Anal. Biochem.* **59**: 583–598 (1974).

[17] Hamilton, W.C., *Statistics in Physical Science*, (Ronald Press Company, New York, 1964).

[18] Ruhe, A. and Wedin, P.A., Algorithms for separable nonlinear least squares problems, *SIAM Rev.* **22**: 318–337 (1980).

[19] Jennrich, R.I. and Sampson, P.F., Application of stepwise regression to nonlinear estimation, *Technometrics* **10**: 63–72 (1968).

[20] Provencher, S.W., CONTIN (Version 2) Users Manual, *Technical Report* EMBL-DA05 (1982).

8. INDEXES

8.1 Important Equations

$$y_k \approx \sum_{j=1}^{N_\lambda} \alpha_j f(\lambda_j, t_k) + \sum_{\mu=1}^{N_\gamma} \gamma_\mu g_\mu(t_k) =: \hat{y}(t_k) \ , \quad k = 1, \ldots, N_y \qquad (3.1-1)$$

$$\mathtt{VAR} = \sum_{k=1}^{N_y} w_k (y_k - \hat{y}_k)^2 = \text{minimum} \qquad (3.2-1)$$

$$f(\lambda, t_k) = \sum_{l=1}^{N_B} \beta_{lk} B_l(\lambda) \qquad (3.2-2)$$

$$y_{kn} \approx \sum_{j=1}^{N_\lambda} \alpha_{jn} f(\lambda_j, t_{kn}) + \sum_{\mu=1}^{N_\gamma} \gamma_{\mu n} g_\mu(t_{kn}) =: \hat{y}(t_{kn}) \ , \qquad (4.1.2-1)$$

$$k = 1, \ldots, N_{y_n} \ , \ \ n = 1, \ldots, N_D$$

$$\mathtt{VAR} = \sum_{n=1}^{N_D} \sum_{k=1}^{N_{y_n}} w_{kn} (y_{kn} - \hat{y}_{kn})^2 \qquad (4.1.2-2)$$

$$f(\lambda, t_{kn}) = \sum_{l=1}^{N_B} \beta_l(t_{kn}) B_l(\lambda) \qquad (4.1.6-1)$$

8.2 Control Variables

Name	Type [1] DIMENSION	FORMAT (2^{nd} card)	Default value(s)	Defined in section
CONVRG	R		5.E-5	4.1.7.3
DOADEX	L(10,2)	20L1	20*F	4.1.6
DOSPL	L(10,2)	20L1	20*T	4.1.6
DOSTRT	L(10,2)	20L1	20*F	4.1.7.4
DOUSIN	L		F	4.1.4
DOUSOU	L(2)		2*F	4.1.5.1
IFORMT	I(70)	1X,70A1	(5E15.6)	3.4.1
IFORMW	I(70)	1X,70A1	(5E15.6)	3.4.1
IFORMY	I(70)	1X,70A1	(5E15.6)	3.4.1
IPLFIT	I(2)		0,0	4.1.5.2
IPLRES	I(2)		0,1	4.1.5.2
IPRINT	I(3,2)	6I4	3*1,3*1	4.1.5.2
IPRITR	I(2)		0,2	4.1.5.2
IRWCAP	I		0	4.1.7.1
IUNIT	I		0	4.1.7.1
IUSER	I(50)		50*0	4.1.1
IWT	I		1	4.1.3.4
LAST	L		T	3.4.1
LINEPG	I		60	3.4.4
LUSER	L(30)		30*F	4.1.1

[1] where I = INTEGER, R = REAL, and L = LOGICAL

Name	Type [1] DIMENSION	FORMAT (2nd card)	Default value(s)	Defined in section
MCONV	I		3	4.1.7.3
METHOD	I		2	4.1.7.5
MIOERR	I		5	4.1.5.2
MTRY	I(10,2)	20I4	10,9*20,20,9*50	4.1.7.4
MXITER	I(2)		20,40	4.1.7.3
NABORT	I		3	4.1.7.3
ND	I		1	4.1.2
NERFIT	I		10	4.1.3.4
NGAM	I		0	4.1.3.3
NL	I(10)		1,2,3,4,5,6,7,8,9,10	3.4.2
NNL	I		1	3.4.2
NZ	I		50	4.1.6
PLMNMX	R(2)		-1.E+5,1.E+5	3.4.3
PNMNMX	R(2)		.02,2.08	3.4.3
PRWT	L		T	4.1.5.2
PRY	L		T	4.1.5.2
RUSER	R(100)		100*0.	4.1.1
SAMET	L		F	4.1.2
SIMULA	L		F	4.1.7.2

[1] where I = INTEGER, R = REAL, and L = LOGICAL

8.3 Other Terms