ornl

OAK
RIDGE
NATIONAL
LABORATORY

UNION
CARBIDE

ORNL/TM-8185
ENDF-323

User's Guide for BAYES: A General-Purpose Computer Code for Fitting a Functional Form to Experimental Data

Nancy M. Larson

OPERATED BY
UNION CARBIDE CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
NTIS price codes—Printed Copy: A15 Microfiche A01

ORNL/TM-8185
ENDF-323

Contract No. W-7405-eng-26

Engineering Physics Division
and Physics Division

User's Guide for BAYES: A General-Purpose Computer Code for Fitting a Functional Form to Experimental Data

Nancy M. Larson
Computer Sciences

Date Published - August 1982

OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37830
operated by
UNION CARBIDE CORPORATION
for the
DEPARTMENT OF ENERGY

TABLE OF CONTENTS

LIST OF TABLES

ACKNOWLEDGMENTS

The author acknowledges contributions from a multitude of people, without whose aid this project might never have been brought to fruition. First and foremost, Francis Perey provided invaluable insight into the correct use of Bayes' method. Reg Gwin, in his struggles to use an early version of BAYES, suggested several modifications to increase the versatility and thus the utility of the code. Other early users of BAYES (David Hetrick, Peter Fu, Kirk Dickens, and Duane Larson) provided both encouragement and inspiration for the examples used in this report. The financial and moral support of the entire ORELA staff is greatly appreciated.

Finally, thanks go to Betty Waddell and Bobbie Roth who patiently typed the several iterations of this report.

ABSTRACT

This report is intended as a user's manual for a general-purpose computer program BAYES to solve "Bayes' equations" for updating parameter values, uncertainties, and correlations. Bayes' equations are derived from Bayes' theorem, using linearity and normality assumptions. The method of solution is described, and details are given for adapting the code for a specific purpose. Numerous examples are given, including problem description and solution method, FØRTRAN coding, and sample input and output. A companion code LEAST, which solves the usual least-squares equations rather than Bayes' equations but which encourages nondiagonal data weighting, is also described.

I. INTRODUCTION

In a recent report [LA80] on a multilevel R-matrix code (SAMMY) for analysis of experimental neutron data, we described the use of Bayes' method for obtaining estimates of parameters of a model by fitting experimental data. In this report we describe a "skeleton" code (BAYES) which can be used to solve Bayes' equations for an arbitrary problem, much as one would use a generalized least-squares solver.

A discussion of the philosophy regarding the use of Bayes' theorem is presented in the earlier report and will not be repeated here. Rather, in Section II we present a straightforward derivation of Bayes' equations, carefully delineating the assumptions required for that derivation. For further details, the reader is referred to the forthcoming article on Bayes' method [PE82].

In Section III we describe the manner in which Bayes' equations are implemented in the computer code BAYES. In Section IV other portions of BAYES are discussed in sufficient detail to permit ready modification for specific applications.

Section V is expected to be the most important section for anyone wishing to employ Bayes' method in his own work. Here we present examples of problems which might be attacked with Bayes' method. A description of a problem is given, followed by a discussion of the various approaches one could take to the problem. The FØRTRAN coding which the user would supply is given for each approach discussed. Also shown are sample input and output.

In Section VI we describe a companion program (LEAST), identical to BAYES except that it is used to solve the least-squares equations rather than Bayes' equations. Applications of LEAST to the examples of Section V are also given in Section VI.

Five appendices complete this report. Appendix A gives those algebraic details which were judged too complex to be included within the text. Appendix B contains an alphabetized table of notation used in BAYES. The FØRTRAN listing of subroutines unique to BAYES is given in Appendix C, of those unique to LEAST in Appendix D, and of the output routines in Appendix E.

II. DERIVATION OF BAYES' EQUATIONS

In this section the formulae implemented in the code BAYES to obtain the values of the parameters and their uncertainties using Bayes' theorem are derived. Bayes' theorem may be written in the form

$$p(P|DX) \propto p(P|X)p(D|PX) \tag{II.1}$$

where

(1) P represents the parameters of whatever physical theory is to be used to describe the data and D represents the new experimental data which the theory is expected to describe.

(2) X represents "background" or "prior" information such as the data from which prior knowledge of the parameters P was derived. X is assumed to be independent of D.

(3) $p(P|DX)$ is the probability for the value of the parameters conditional upon the new data D and is what we seek. It is conventional to call $p(P|DX)$ the posterior probability. When P represents several parameters, $p(P|DX)$ is a joint probability density function (joint pdf). The expectation values of P times $p(P|DX)$ are taken as the new estimates for the parameters; the associated covariance matrix gives us a measure of how well the parameters are determined and of the parameters' inter-dependencies.

(4) $p(D|PX)$ is the probability density function for observing the data D given that the parameters P are correct. It is a function of the parameters P of the model and is proportional to the likelihood function of the data D.

(5) $p(P|X)$ is the joint pdf for the value of the parameters P of the model, prior to consideration of the new data D; it is known as the prior joint pdf. The expectation values of P times $p(P|X)$ are the prior estimates for the values of the parameters; the associated covariance matrix gives a measure of how well the parameters are known before consideration of the new data.

(6) The constant of proportionality in Eq. (II.1) can be determined from the normalization condition.

Let $P = \{P_k\}$ for k = 1 to K be the set of all parameters of the theoretical model to be considered. The joint pdf $p(P|X)$ is assumed to be a joint normal pdf having as expectation value the vector $\bar{P}$ and the covariance matrix M. Under this assumption the pdf may be written

$$p(P|X) \propto \exp[\ -1/2\ (P-\bar{P})^t\ M^{-1}(P-\bar{P})]\ , \tag{II.2}$$

where the superscript t denotes the transpose.

The experimental data is represented by a data vector D whose components D_i are the L data points. The experimental conditions are assumed to be such that the data D (i.e. the D_i's) have a joint normal distribution with mean T=T(P) and covariance matrix V. The likelihood function is then

$$p(D|PX) \propto \exp[\ -1/2\ (D-T)^t\ V^{-1}(D-T)]\ . \tag{II.3}$$

In this equation, the covariance matrix V represents not only the experimental "errors" of the data, but also any theoretical "errors" resulting from approximations used in calculating T. Obviously V need not be diagonal.

Combining Eqs. (II.1), (II.2), and (II.3) gives an expression for the pdf of P after consideration of new data D (i.e., for $p(P|DX)$), expressed in terms of the "true" value T. What is needed, however, is an expression for $p(P|DX)$ expressed in terms of the parameters P. This is obtained formally by considering T a function of P, performing a Taylor expansion about $\bar{P}$ (the expectation value of $p(P|X)$), and keeping only the linear terms:

$$T(P) \simeq \bar{T} + G(P-\bar{P}) \quad , \tag{II.4}$$

where $\bar{T}$ is equal to $T(\bar{P})$. The elements of G are the partial derivatives of T_n with respect to the parameters P_k, evaluated at $P = \bar{P}$:

$$G_{nk} = \left. \frac{\partial T_n}{\partial P_k} \right|_{P = \bar{P}} \qquad \begin{array}{l} \text{for } n = 1,2,\ldots,L \\ \text{and } k = 1,2,\ldots,K \end{array} \tag{II.5}$$

Since T is a vector of dimension L (equal to the number of data points), and P is a vector of dimension K (equal to the number of parameters), this "sensitivity matrix" G is of dimension $L \times K$.

Substituting Eq. (II.4) into Eq. (II.3) and using Eq. (II.2), we obtain for the posterior joint pdf (Eq. (II.1)):

$$p(P|DX) \propto \exp[-\ 1/2\ \{(P-\bar{P})^t\ M^{-1}(P-\bar{P})\ +$$

$$(D-\bar{T}-G(P-\bar{P}))^t\ V^{-1}(D-\bar{T}-G(P-\bar{P}))\}]. \tag{II.6}$$

Because of the three basic assumptions we have made, i.e.,

i) the prior joint pdf is a joint normal,

ii) the likelihood function is a joint normal, and

iii) the true value is a linear function of the parameters,

the posterior joint pdf is also a joint normal. Denoting its expectation value by $\bar{P}'$ and its covariance matrix by M', we may write:

$$p(P|DX) \propto \exp[\, -1/2(P-\bar{P}')^t \, M'^{-1} \, (P-\bar{P}')]. \qquad \text{(II.7)}$$

As shown in Appendix A, equating the linear and quadratic terms of the exponents in Eqs. (II.6) and (II.7) yields our final results, hereafter referred to as Bayes' equations:

$$\bar{P}' - \bar{P} = M\, G^t\, (N+V)^{-1}\, (D-\bar{T}) \quad , \qquad \text{(II.8)}$$

$$M - M' = M\, G^t\, (N+V)^{-1}\, G\, M \, , \qquad \text{(II.9)}$$

where the $L \times L$ matrix N is defined as

$$N = G\, M\, G^t. \qquad \text{(II.10)}$$

The matrix N is the covariance matrix of the joint pdf for the true value of the data based upon our prior pdf for the value of the parameters.

In the limit where the matrix M is diagonal and its elements tend to infinity, Bayes' equations become the familiar least-squares equations. Algebraic details are given in Appendix A.

Because the linearity condition (Eq. (II.4)) may be only approximately correct, it is necessary to alter Bayes' equations slightly to permit iteration to an accurate solution. (Details are given in Appendix A.) It is the iterative form of Bayes' equations which is implemented in the code BAYES.

Finally, we note that the derivation of Eqs. (II.8) through (II.10) which we present in Appendix A is not the only possible derivation. Alternative derivations can be found in [MA80] and [GA73].

III. IMPLEMENTATION OF BAYES' EQUATIONS

1. Solving the Equations

Implementation of Bayes' equations is straightforward. From Appendix A, the iterative form of these equations is

$$\bar{P}^{(n+1)} = \bar{P} + MG^{(n)t} \, (N^{(n)}+V)^{-1} \, (D-\bar{T}^{(n)} - G^{(n)}(\bar{P}-\bar{P}^{(n)})) \qquad \text{(III.1)}$$

and

$$M^{(n+1)} = M - MG^{(n)t} \, (N^{(n)}+V)^{-1} \, G^{(n)}M, \qquad \text{(III.2)}$$

where

$$N^{(n)} = G^{(n)}MG^{(n)t} \, . \qquad \text{(III.3)}$$

Solving Eqs. (III.1) and (III.2) is equivalent to solving

$$AX = Y \qquad \text{(III.4)}$$

K+1 times (where K is the number of parameters for the problem), with A the $L \times L$ symmetric matrix N+V (where L is the number of data points), and Y a column matrix equal to $(D-\bar{T}^{(n)}-G^{(n)}(\bar{P}-\bar{P}^{(n)}))$ in Eq. (III.1) or equal to each of the K columns of the rectangular matrix $G^{(n)}M$ in Eq. (III.2). The results for X are then multiplied by $MG^{(n)t}$ for use in Eqs. (III.1) and (III.2).

In order to ensure reasonable accuracy in the solution of Eq. (III.4), we perform a scaling operation on matrix A prior to solution of the equation. Note that Eq. (III.4) is mathematically equivalent to the equation

$$A'X' = Y' \qquad \text{(III.5)}$$

where $A'=CAC$, $X'=C^{-1}X$, and $Y'=CY$, with C any $L \times L$ nonsingular matrix. In particular, we choose C to be the diagonal matrix whose elements are the inverse of the diagonal elements of A. The matrix A' then has 1.0 everywhere on the diagonal, and the numerical solution of Eq. (III.5) is likely to be more accurate than that of Eq. (III.4). A detailed discussion of scaling as preparation for solution of matrix equations is given in Section I.7 of the Linpack Users' Guide [DØ79].

Solution of Eq. (III.5) is obtained in two steps. First matrix A' is factorized into the form

$$A' = UBU^t \tag{III.6}$$

where B is a block-diagonal matrix and U is the product of elementary unit triangular and permutation matrices. The inverses of U and B are then immediately available, and the solution X' is simply

$$X' = (U^{-1})^t \; B^{-1} \; U^{-1}Y'. \tag{III.7}$$

LINPACK subroutine DPPCØ is used to factorize A' according to Eq. (III.6). This need be done only one time, and the (K+1) solutions X' are then found using LINPACK routine DPPSL.

In program BAYES, subroutine NEWPAR oversees the operations outlined above for obtaining updated parameter values and parameter covariance matrix elements.

Two other functions, not directly related to solution of Bayes' equations, are performed in subroutine NEWPAR. The first is to test for convergence, the second to generate χ^2 and/or weighted residuals.

Convergence of the solution is tested as follows: the new value of each parameter is compared to its value at the previous iteration. When all NPAR parameters agree to within a specified tolerance, iteration ceases. Specifically, let t be the tolerance or "convergence fraction" as specified in the teletype input (see Section V, Examples); iteration ceases when

$$\left| \bar{P}^{(n+1)} - \bar{P}^{(n)} \right| < t \left| \bar{P}^{(n)} \right| \tag{III.8}$$

for every varied parameter.

Both least-squares and "Bayesian" χ^2 and weighted residuals may be generated in NEWPAR. The least-squares residuals are defined as

$$R_{LS} = V^{-1}(D - \bar{T}^{(n)}) \tag{III.9}$$

and χ^2 as

$$\chi^2_{LS} = (D - \bar{T}^{(n)})^t \; V^{-1} \; (D - \bar{T}^{(n)}). \tag{III.10}$$

The Bayesian weighted residuals are defined by

$$R_B = (N^{(n)} + V)^{-1}(D - \bar{T}^{(n)} - G^{(n)}(\bar{P} - \bar{P}^{(n)})) \tag{III.11}$$

and χ^2 by

$$\chi^2_B = (D - \bar{T}^{(n)} - G^{(n)} \; (\bar{P} - \bar{P}^{(n)}))^t \; (N^{(n)} + V)^{-1}$$

$$\times (D - \bar{T}^{(n)} - G^{(n)}(\bar{P} - \bar{P}^{(n)})) \quad . \tag{III.12}$$

IV. DETAILS OF THE COMPUTER CODE

1. General Comments

Program BAYES will likely be modified by each user to suit his special purposes. For that reason we have endeavored to make the FØRTRAN coding as readable as possible, perhaps at the expense of computer efficiency. All arrays required by BAYES are of fixed size; error messages appear at execution time when array bounds are exceeded. (Dynamic allocation of array storage is employed in version ZBAYES, described in subsection 6 below.) Comment cards abound throughout the program. Double precision is used in BAYES; though it is not strictly necessary, experience has shown that this is the safer course, especially since numerical "derivatives" are rarely sufficiently accurate in single precision.

For a simple problem, the user must supply only three subroutines: one to initialize parameters and parameter covariances, a second to specify the experimental data and data covariances, and a third to calculate theoretical values. An optional fourth subroutine would generate partial derivatives, though BAYES is capable of generating numerical differences as approximations to the partial derivatives if the user so desires.

Input formats are controlled by the individual user via the subroutines discussed above, which are described in more detail below. The code BAYES, however, handles many of the bookkeeping problems internally:
(1) To vary only selected members of a parameter list, and hold the others constant, the user need only set a flag for the varied parameters.
(2) Though covariance matrices are stored in "packed" form to minimize storage, the user is asked to specify the matrices in the customary two-dimensional form.

All output provided by BAYES is done in routines whose names begin with the letters ØUT. Often a user will not need all possible kinds of output, in which case he may simply suppress the call to that routine by preparing a file called BAYES.ØUT. This file contains one card per suppressed subroutine, with the subroutine's name in columns 1 through 6. For example, listing the values of the partial derivatives will be needed only during the "debugging" stage, so in normal operation file BAYES.ØUT contains a line with "ØUTG (blank) (blank)" in columns 1 through 6. See, e.g., Example 1 in Section V.

Output routines may require modification for some problems. See, e.g., Example 2 in Section V, for which the independent variable is not energy but rather alphanumeric information. Appropriate output formats must be provided by the user for this type of variable.

The one other modification which will often be required is in array dimensions. Such changes of dimension should be made with great care: (1) The appropriate limit NDATMX (maximum number of data points), NPARMX (maximum number of varied parameters), or NNPARM (maximum total number of parameters) must be changed in the DATA statement following the CØMMØN statements in the main program. (2) Dimensions must be changed in CØMMØN/DAT/,/PAR/,/BØTH/, and /YRPAR/ in every subroutine in the entire program. The dimension of VARDAT and EN in CØMMØN/DAT/ should be set equal to ((NDATMX*(NDATMX+1))/2), and the dimension of VARPAR and VARNEW in CØMMØN/PAR/ should be set to ((NPARMX*(NPARMX+1))/2). (3) If NPARMX is changed, the "DIMENSIØN PARVAR" statement in subroutine PPARAM must be changed. (4) If NDATMX is changed, the "DIMENSIØN DATCØV" statement in subroutines SETDAT and FIXV must be changed.

Array sizes in BAYES are currently set at NDATMX=51, NPARMX=25, and NNPARM=26. The error message which appears if these limits are exceeded specifies which CØMMØNS need to be modifed.

Subsections 2 through 5 of this section contain descriptions of the four different categories of subroutines which together comprise the BAYES code. Routines in each category are stored in separated files on disk NML2, Project-Programmer Number (PPN) [100,1006], on the ORELA PDP-10 computer, and can be copied from these to other PPN's. Users not on that computer should contact the Radiation Shielding Information Center [RA82] for access to the code. It is expected (though not guaranteed) that conversion of BAYES (or LEAST) for use on other computer systems should not be difficult, since little PDP-10-specific FORTRAN has been employed.

Subsection 6 describes version ZBAYES, in which arrays are not of fixed dimensions.

2. User-Supplied Subroutines

Three routines must be provided by the users for every problem. The first is subroutine SETPAR, which initializes parameters and parameter covariances, and flags those parameters which are to be varied. (Note: if no parameter is flagged, BAYES assumes all parameters are to be varied.) Users unfamiliar with Bayes' method may question what are appropriate values for the covariance matrix elements; often a reasonable procedure is to set off-diagonal elements equal to zero and diagonal elements equal to the square of the uncertainty on the parameter. For example, if previous experiments have shown that the parameter cannot vary by more than 10%, then the diagonal element is $(0.1 \times P)^2$. If there is no prior evidence constraining the parameter to a certain size, the diagonal element may be initialized at $(1.0 \times P)^2$ or even greater. Strictly speaking, a more correct approach in this case would be to use the least-squares method; however, that is not practical if prior information exists for some parameters but not for others, or if multiple data sets are to be analyzed sequentially.

The second user-supplied routine is subroutine SETDAT, which specifies the experimental data, the values of the independent variable or variables labeling each data point, and the data covariance matrix elements. For guidance in setting the values of data covariances, the user is referred to the examples in Section V. Note that in many cases it is not appropriate to assume that the data covariance matrix is diagonal.

The third routine is function THEØ(KDAT), which generates the theoretical value to which experimental data point KDAT is to be compared.

For many problems, BAYES can generate numerical differences to approximate the partial derivative of the theoretical value with respect to each of the parameters. However, in most instances the user should provide analytical derivatives in function DERIV, since the numerical difference option consumes considerable computer time. Where possible, the numerical difference option should be used for debug purposes, to test both algebra and coding in function DERIV.

FØRTRAN listings of the four routines SETPAR, SETDAT, THEØ, and DERIV are given in Section V for each example covered in that section; Table IV.1 summarizes the information the user must supply in the four routines. In modifying the routines for his problem, the user should take care to change only the executable statements and not CØMMØN blocks or EQUIVALENCE statements. (It may, of course, be necessary to change array dimensions if limits are exceeded.) If the automatic numerical difference option is to be used exclusively, function DERIV can be a "dummy" routine of the form

```
DØUBLE PRECISIØN FUNCTIØN DERIV
RETURN
END
```

Table IV.1. Description of the Four Subprograms Which a User Must Supply.

Routine	Variable or Array to be Initialized	Meaning
Subroutine SETPAR	NPARAM	total number of parameters
	PARAM(i), for i=1 to NPARAM	$\bar{P}_i$ for all i
	IFPAR(i), for i=1 to NPARAM	1 if parameter number i is to be varied, 0 otherwise
	PARCØV(k,i) for k and i=1 to NPARAM	M_{ki} (only $k \leq i$ need be specified, since the array is symmetric)
Subroutine SETDAT	NDAT	number of data points
	E(i), for i=1 to NDAT	the first independent variable (often energy)
	E2(i) for i=1 to NDAT	the second independent variable, if needed
	DATA(i) for i=1 to NDAT	experimental data
	DATCØV(k,i) for k and i=1 to NDAT	V_{ki} (only $k \leq i$ need be specified, since the array is symmetric); Note that the uncertainty on data point i is $\sqrt{V_{ii}}$

Table IV.1. (Continued)

Routine	Variable or Array to be Initialized	Meaning
Double Precision Function THEØ	THEØ(KDAT)	$\bar{T}^{(n)}$(KDAT) =theory for data point KDAT, evaluated at parameter values PARAM (i.e., $\bar{P}^{(n)}$)
Double Precision Function DERIV	DERIV(KDAT,KPAR)	$G^n_{ij} = \frac{\partial T_i}{\partial P_j}$ i.e., the partial derivative of the theory with respect to parameter number j=KPAR, for data point i=KDAT, evaluated at parameter values PARAM (i.e. $\bar{P}^{(n)}$)

3. Output Routines

Output from BAYES is generated only in routines whose names begin with the letters ØUT. These routines are listed in Table IV.2 and stored as file ØUT.FØR on the ORELA PDP-10 computer. Any of these routines may be modified by the user to provide output more suitable to his needs. A summary of what is printed in each routine is given in Table IV.2. FØRTRAN listing is provided in Appendix E.

Unit 6 is the logical unit number for output from BAYES. On the PDP-10, this causes output to be printed in a file having extension LPT, which can then be typed or QUEUED to the line printer. When the file BAYES.ØUT does not specify suppression of the call to subroutine ØUTGMG,the extensive output from that routine appears in file FØR28.DAT.

TABLE IV.2. Description of Output Subroutines

Subroutine Name	Name of array printed in that routine	Meaning
ØUTALL	(None)	(read file BAYES.ØUT to learn which output is to be suppressed)
ØUTCHB, ØUTCHL	CHI	χ^2 {Bayes version is given in (Eq. III.12); Least-squares version in (Eq. III.10)}
ØUTDAT	E	energies (independent variables)
	DATA	experimental data
	VARDAT	covariance matrix (output as standard deviation plus correlation matrix)
ØUTG	E	energies
	G	partial derivatives
ØUTGMG	GMG	(for debug only)
ØUTPAR	PARM	initial values for varied parameters
	VARPAR	initial values for parameter covariance matrix
ØUTP1	PØLD	old values of parameters
	PARM	updated values of parameters
ØUTP2	PØLD	old values of parameters
	PARM	updated values of parameters
	VARNEW	updated covariance matrix
ØUTREB, ØUTREL	E	energies
	DUM	weighted residuals {Bayes version from (Eq. III.11); Least-squares version from (Eq. III.9)}
ØUTTH	E	energies
	DATA	data
	TH	theoretical values

TABLE IV.2. (Continued)

Subroutine Name	Name of array printed in that routine	Meaning
ØUTV	V	covariance matrix V_{ki}
ØUTYV	PARCØV	covariance matrix M_{ki}
ØUTYPR	PARAM PARCØV	complete set of parameters uncertainties and correlations (i.e., covariance matrix)

4. Black-Box Routines

The method of solution of Bayes' equations (described in Section III) requires a significant amount of matrix manipulation. BAYES uses the LINPACK [DØ79] routines DPPCØ, DPPSL, and others called by those two, to perform these manipulations. These routines may safely be treated as a "black box" and therefore are not listed in this report. They are available on the PDP-10 in file DPPXX.FØR.

Automatic generation of numerical differences as approximations to analytic derivatives is accomplished by a call to subroutine DO4NML, which is a double-precision modification of the Numerical Algorithms Group routine DO4AAF [NAG]. This routine and those called by it are available on the PDP-10 in file DO4NML.FØR.

5. Bayes Routines

The remaining subroutines of the code BAYES are stored in file BAYES.FØR on the PDP-10 and listed in Appendix C. This includes all routines in solid boxes in the tree diagram shown in Fig. IV.1.

The main program functions as an overseer, calling appropriate subroutines to perform the actual calculations. The only modification a user should make in the main program is to change dimensions of the arrays.

Other subroutines included in this segment will not likely require modification by the casual user. Briefly, these routines are:

PPARAM, which performs bookkeeping operations to convert between the complete parameter set and the varied parameter set.

FIXV, which rearranges storage of data covariance matrix.

THEØRY, which oversees evaluation of theoretical values and partial derivatives.

FUN, an auxiliary program required by the automatic numerical derivative routine.

NEWPAR, which, along with SCALE, MUL, and MUL2, solves Bayes' equations and generates weighted residuals and X^2 values.

UPDATE, which updates parameter values in the complete parameter set.

RESTRT, which reinitializes arrays in order to analyze another data set.

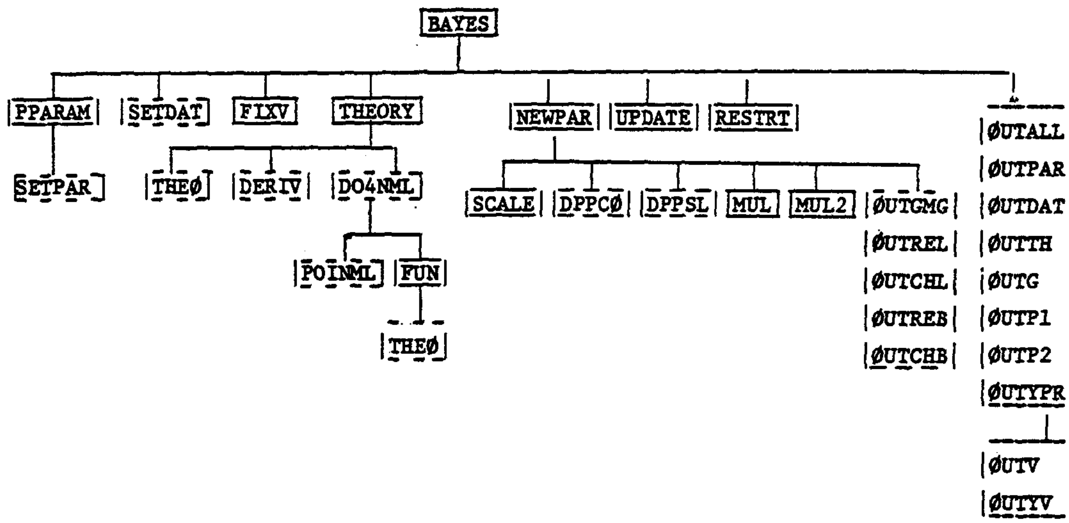

Fig. IV.1. Tree Diagram of Subroutines in BAYES.

6. Variable-Dimensioned Arrays

Experienced computer programmers may find fixed array dimensions burdensome; therefore, an alternative version ZBAYES has been prepared. ZBAYES is identical to BAYES, except that array storage is allocated as it is needed, using calls to function IDIMEN to fix REAL*8 arrays, and calls to JDIMEN to fix INTEGER arrays. Thus, for example, the statement

```
IDATA = IDIMEN(NDAT)
```

causes NDAT words to be set aside in array A in CØMMØN/EXPAND/NREAL,INTEG,A,IA. Here A is a double precision array of dimension NREAL, and IA is an integer array of dimension INTEG. Calls to a subroutine which uses array DATA are then written as

```
CALL SUB(A(IDATA)).
```

Once DATA is no longer needed, its space may be released via

```
I = IDIMEN(-IDATA).
```

It should be noted that this also releases everything allocated after DATA.

The dimension of array A is specified by NREAL, and of IA by INTEG. A warning message is printed if these limits are exceeded, in which case the user needs simply to increase the size and recompile the main routine.

No listing of ZBAYES is provided within this report. ORELA PDP-10 users may obtain the necessary files from NML2: [100,1006]; filenames are identical to those specified for BAYES, with prefix Z appended. Other users should contact RSIC [RA82]. Conversion of ZBAYES for use on other computers should involve no major difficulties.

V. EXAMPLES

Perhaps the clearest way to illustrate how a skeleton program such as BAYES might be used is by providing a number of examples. In this section we describe five different problems, identify one or more possible solutions to each problem, and list the FØRTRAN coding used to obtain the solutions. Sample input and output are provided for each case.

PDP-10 filenames are given for both data and user-supplied subroutines, for each of the four examples. ORELA PDP-10 users can access these files from disk NML2, project-programmer number [100,1006]. Output files are not saved, since they can be readily regenerated and compared to the listings provided here. Output is on unit 6, and will appear in a file named ???.LPT on the ORELA PDP-10.

Example 1. Fit a Functional Form to Experimental Data

The Problem:

The analytic expression f(E,P), where E is the independent variable and P is one or more parameters, is expected to provide a reasonable fit to a set of experimental data D. Find the parameters $\bar{P}'$, and the corresponding covariance matrix M', which provide the best fit.

The Solution:

The solution is straightforward. Set

$$\bar{T}_i = f(E_i,\bar{P}) \qquad (V.1.1)$$

and

$$G_{ij} = \left.\frac{\partial f}{\partial P_j}\right|_{E=E_i,\ P=\bar{P}} \qquad (V.1.2)$$

and solve Bayes' equations for $\bar{P}'$ and M'.

Specific Case:

Let the function f(E,P) be given by

$$f(E,\ P) = (P_1\ E)^2 + P_2 E + P_3. \qquad (V.1.3)$$

Initial values for the parameters are assumed to be known to within 10%, from some "previous experiment." The current "experiment " provides 21 (artificially generated) uncorrelated data points, as listed in Table V.1.1 (the PDP-10 file EX1.DAT). The PDP-10 file EX1.F4 shown in Table V.1.2 contains the user-supplied subroutines for this problem. Appropriate commands to execute this program are given in Table V.1.3, along with user responses to program prompts. The corresponding output file is given in Table V.1.4.

Teletype input for using the automatic numerical derivative option is shown in Table V.1.5, with corresponding output in Table V.1.6. In this case we have suppressed most of the output by generating the file BAYES.ØUT as shown in Table V.1.7.

Table V.1.1. ØRELA PDP-10 File EX1.DAT,
Containing Data Required for Example 1.

Energy	Data	Relative Error
10.00	154.00	0.01
11.00	151.90	0.01
12.00	152.00	0.01
13.00	154.30	0.01
14.00	158.80	0.01
15.00	165.50	0.01
16.00	174.40	0.01
17.00	185.50	0.01
18.00	198.80	0.01
19.00	214.30	0.01
20.00	232.00	0.01
21.00	251.90	0.01
22.00	274.00	0.01
23.00	298.30	0.01
24.00	324.80	0.01
25.00	353.50	0.01
26.00	384.40	0.01
27.00	417.50	0.01
28.00	452.80	0.01
29.00	490.30	0.01
30.00	530.00	0.01

Table V.1.2. User-Supplied Subroutines for Example 1, Found on File EX1.F4 on the ØRELA PDP-10.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARAM, PARCOV, AND IFPAR (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C        NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 3
      WRITE (5,99999)
      READ (5,99998) (PARAM(I),I=1,NPARAM)
C
      DO 40 I=1,NPARAM
           DO 30 L=1,I
                PARCOV(L,I) = 0.D0
                IF (L.EQ.I) PARCOV(L,I) = .01D0*PARAM(I)**2
   30      CONTINUE
   40 CONTINUE
      RETURN
C
99999 FORMAT (33H INITIAL GUESSES FOR PARAMETERS? $)
99998 FORMAT (3F)
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
C        NML, JANUARY 1981
```

Table V.1.2. (Continued)

```
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME)
C
C *** READ FILE 24 TO COUNT THE NUMBER OF DATA POINTS (NDAT)
      N = 0
   10 READ (24,99997,END=20) EE, D, ERR
      IF (EE.LE.0.D0) GO TO 20
      N = N + 1
      GO TO 10
   20 CONTINUE
      NDAT = N
      REWIND 24
C
C
C *** INITIALIZE DATA COVARIANCE MATRIX DATCOV
      DO 40 I=1,NDAT
           DO 30 L=1,I
                DATCOV(L,I) = 0.D0
   30      CONTINUE
   40 CONTINUE
C
C *** READ FILE 24 TO OBTAIN ENERGY, DATA, AND UNCERTAINTIES,
C ***   SET COVARIANCE MATRIX
      DO 50 I=1,NDAT
           READ (24,99997) EE, D, ERR
           E(I) = EE
           DATA(I) = D
           DATCOV(I,I) = (ERR*D)**2
   50 CONTINUE
      CLOSE (UNIT=24)
      RETURN
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (3F20.10)
      END
```

Table V.1.2. (Continued)

```
C
C
C
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C       NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      THEO = (PARAM(1)**2*E(KDAT)+PARAM(2))*E(KDAT) + PARAM(3)
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR ALL PARAMETERS,
C ***            AT ENERGY E(KDAT)
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      GO TO (10, 20, 30), KPAR
   10 CONTINUE
      GG = 2.*PARAM(1)*E(KDAT)**2
      GO TO 40
   20 CONTINUE
      GG = E(KDAT)
      GO TO 40
   30 CONTINUE
      GG = 1.D0
   40 CONTINUE
      DERIV = GG
      RETURN
      END
```

Table V.1.3. Teletype Commands and User Responses to Program Prompts for Example 1. Underlined portions are supplied by the user; a downward arrow indicates carriage return.

```
.EX BAYES,EX1,SHOW,DPPXX,DO4NML↓
HOW MANY ITERATIONS? WHAT IS CONVERGENCE FRACTION?  5,.001↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES?  N↓
INITIAL GUESS FOR PARAMETERS?  1.,-25.,300.↓
WHAT'S NAME OF THE DATA FILE?  EX1.DAT↓
WHAT'S NAME OF THE DATA FILE?  ↓
STOP
```

Table V.1.4. Output Resulting from Use of the Commands Shown in Table V.1.3.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

******INPUT PARAMETER VALUES

        PARAMETER       UNCERTAINTY
  1      1.000000       0.100000
  2    -25.000000       2.500000
  3    300.000000      30.000000

****** INPUT DATA VALUES

       DATA POINT         VALUE       UNCERTAINTY
  1     10.000000      154.000000       1.540000
  2     11.000000      151.900000       1.519000
  3     12.000000      152.000000       1.520000
  4     13.000000      154.299999       1.543000
  5     14.000000      158.799999       1.588000
  6     15.000000      165.500000       1.655000
  7     16.000000      174.400000       1.744000
  8     17.000000      185.500000       1.855000
  9     18.000000      198.799999       1.988000
 10     19.000000      214.299999       2.143000
 11     20.000000      232.000000       2.320000
 12     21.000000      251.900000       2.519000
 13     22.000000      274.000000       2.740000
 14     23.000000      298.299999       2.983000
 15     24.000000      324.799999       3.248000
 16     25.000000      353.500000       3.535000
 17     26.000000      384.400002       3.844000
 18     27.000000      417.500000       4.175000
 19     28.000000      452.799999       4.528000
 20     29.000000      490.299999       4.903000
 21     30.000000      530.000000       5.300000
```

Table V.1.4. (Continued)

***** THEORETICAL CALCULATION

	ENERGY	DATA	THEORY
1	10.000000	154.000000	150.000000
2	11.000000	151.900000	146.000000
3	12.000000	152.000000	144.000000
4	13.000000	154.299999	144.000000
5	14.000000	158.799999	146.000000
6	15.000000	165.500000	150.000000
7	16.000000	174.400000	156.000000
8	17.000000	185.500000	164.000000
9	18.000000	198.799999	174.000000
10	19.000000	214.299999	186.000000
11	20.000000	232.000000	200.000000
12	21.000000	251.900000	216.000000
13	22.000000	274.000000	234.000000
14	23.000000	298.299999	254.000000
15	24.000000	324.799999	276.000000
16	25.000000	353.500000	300.000000
17	26.000000	384.400002	326.000000
18	27.000000	417.500000	354.000000
19	28.000000	452.799999	384.000000
20	29.000000	490.299999	416.000000
21	30.000000	530.000000	450.000000

****** PARTIAL DERIVATIVES

	ENERGY	1	2	3
1	1.000000D+01	2.00000D+02	1.00000D+01	1.00000D+00
2	1.100000D+01	2.42000D+02	1.10000D+01	1.00000D+00
3	1.200000D+01	2.88000D+02	1.20000D+01	1.00000D+00
4	1.300000D+01	3.38000D+02	1.30000D+01	1.00000D+00
5	1.400000D+01	3.92000D+02	1.40000D+01	1.00000D+00
6	1.500000D+01	4.50000D+02	1.50000D+01	1.00000D+00
7	1.600000D+01	5.12000D+02	1.60000D+01	1.00000D+00
8	1.700000D+01	5.78000D+02	1.70000D+01	1.00000D+00
9	1.800000D+01	6.48000D+02	1.80000D+01	1.00000D+00
10	1.900000D+01	7.22000D+02	1.90000D+01	1.00000D+00
11	2.000000D+01	8.00000D+02	2.00000D+01	1.00000D+00
12	2.100000D+01	8.82000D+02	2.10000D+01	1.00000D+00
13	2.200000D+01	9.68000D+02	2.20000D+01	1.00000D+00
14	2.300000D+01	1.05800D+03	2.30000D+01	1.00000D+00
15	2.400000D+01	1.15200D+03	2.40000D+01	1.00000D+00
16	2.500000D+01	1.25000D+03	2.50000D+01	1.00000D+00
17	2.600000D+01	1.35200D+03	2.60000D+01	1.00000D+00
18	2.700000D+01	1.45800D+03	2.70000D+01	1.00000D+00
19	2.800000D+01	1.56800D+03	2.80000D+01	1.00000D+00
20	2.900000D+01	1.68200D+03	2.90000D+01	1.00000D+00
21	3.000000D+01	1.80000D+03	3.00000D+01	1.00000D+00

Table V.1.4. (Continued)

***** LEAST-SQUARES WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL		ENERGY	RESIDUAL
(1)	10.0000	1.68663D+00	(13)	22.0000	5.32793D+00
(2)	11.0000	2.55703D+00	(14)	23.0000	4.97849D+00
(3)	12.0000	3.46260D+00	(15)	24.0000	4.62581D+00
(4)	13.0000	4.32619D+00	(16)	25.0000	4.28129D+00
(5)	14.0000	5.07585D+00	(17)	26.0000	3.95227D+00
(6)	15.0000	5.65895D+00	(18)	27.0000	3.64301D+00
(7)	16.0000	6.04957D+00	(19)	28.0000	3.35564D+00
(8)	17.0000	6.24814D+00	(20)	29.0000	3.09076D+00
(9)	18.0000	6.27507D+00	(21)	30.0000	2.84799D+00
(10)	19.0000	6.16229D+00			
(11)	20.0000	5.94530D+00			
(12)	21.0000	5.65768D+00			

***** LEAST-SQUARES CHI SQUARED AT FORMER VALUES OF PARAMETERS IS 3254.747390

CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS 180.819300

***** BAYESIAN WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL		ENERGY	RESIDUAL
(1)	10.0000	8.73294D-03	(13)	22.0000	-8.20982D-04
(2)	11.0000	4.92541D-03	(14)	23.0000	-6.13340D-07
(3)	12.0000	1.43318D-03	(15)	24.0000	7.05770D-04
(4)	13.0000	-1.44963D-03	(16)	25.0000	1.29554D-03
(5)	14.0000	-3.53766D-03	(17)	26.0000	1.77486D-03
(6)	15.0000	-4.78199D-03	(18)	27.0000	2.15435D-03
(7)	16.0000	-5.25522D-03	(19)	28.0000	2.44699D-03
(8)	17.0000	-5.10810D-03	(20)	29.0000	2.66569D-03
(9)	18.0000	-4.52395D-03	(21)	30.0000	2.82256D-03
(10)	19.0000	-3.67749D-03			
(11)	20.0000	-2.71351D-03			
(12)	21.0000	-1.73841D-03			

***** BAYESIAN CHI SQUARED AT FORMER VALUES OF PARAMETERS IS 0.272653

CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS 0.015147

Table V.1.4. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	1.000000	1.049677
2	-25.000000	-25.177087
3	300.000000	295.814766

***** THEORETICAL CALCULATION

	ENERGY	DATA	THEORY
1	10.000000	154.000000	154.226069
2	11.000000	151.900000	152.187239
3	12.000000	152.000000	152.352052
4	13.000000	154.299999	154.720509
5	14.000000	158.799999	159.292610
6	15.000000	165.500000	166.068354
7	16.000000	174.400000	175.047741
8	17.000000	185.500000	186.230772
9	18.000000	198.799999	199.617447
10	19.000000	214.299999	215.207765
11	20.000000	232.000000	233.001726
12	21.000000	251.900000	252.999331
13	22.000000	274.000000	275.200580
14	23.000000	298.299999	299.605472
15	24.000000	324.799999	326.214008
16	25.000000	353.500000	355.026187
17	26.000000	384.400002	386.042010
18	27.000000	417.500000	419.261476
19	28.000000	452.799999	454.684586
20	29.000000	490.299999	492.311340
21	30.000000	530.000000	532.141737

Table V.1.4. (Continued)

****** PARTIAL DERIVATIVES

	ENERGY	1	2	3
1	1.000000D+01	2.09935D+02	1.00000D+01	1.00000D+00
2	1.100000D+01	2.54022D+02	1.10000D+01	1.00000D+00
3	1.200000D+01	3.02307D+02	1.20000D+01	1.00000D+00
4	1.300000D+01	3.54791D+02	1.30000D+01	1.00000D+00
5	1.400000D+01	4.11473D+02	1.40000D+01	1.00000D+00
6	1.500000D+01	4.72355D+02	1.50000D+01	1.00000D+00
7	1.600000D+01	5.37435D+02	1.60000D+01	1.00000D+00
8	1.700000D+01	6.06713D+02	1.70000D+01	1.00000D+00
9	1.800000D+01	6.80191D+02	1.80000D+01	1.00000D+00
10	1.900000D+01	7.57867D+02	1.90000D+01	1.00000D+00
11	2.000000D+01	8.39742D+02	2.00000D+01	1.00000D+00
12	2.100000D+01	9.25815D+02	2.10000D+01	1.00000D+00
13	2.200000D+01	1.01609D+03	2.20000D+01	1.00000D+00
14	2.300000D+01	1.11056D+03	2.30000D+01	1.00000D+00
15	2.400000D+01	1.20923D+03	2.40000D+01	1.00000D+00
16	2.500000D+01	1.31210D+03	2.50000D+01	1.00000D+00
17	2.600000D+01	1.41916D+03	2.60000D+01	1.00000D+00
18	2.700000D+01	1.53043D+03	2.70000D+01	1.00000D+00
19	2.800000D+01	1.64589D+03	2.80000D+01	1.00000D+00
20	2.900000D+01	1.76556D+03	2.90000D+01	1.00000D+00
21	3.000000D+01	1.88942D+03	3.00000D+01	1.00000D+00

***** LEAST-SQUARES WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL
(1)	10.0000	-9.53235D-02
(2)	11.0000	-1.24488D-01
(3)	12.0000	-1.52377D-01
(4)	13.0000	-1.76622D-01
(5)	14.0000	-1.95345D-01
(6)	15.0000	-2.07502D-01
(7)	16.0000	-2.12965D-01
(8)	17.0000	-2.12370D-01
(9)	18.0000	-2.06836D-01
(10)	19.0000	-1.97665D-01
(11)	20.0000	-1.86111D-01
(12)	21.0000	-1.73250D-01

	ENERGY	RESIDUAL
(13)	22.0000	-1.59915D-01
(14)	23.0000	-1.46711D-01
(15)	24.0000	-1.34036D-01
(16)	25.0000	-1.22132D-01
(17)	26.0000	-1.11124D-01
(18)	27.0000	-1.01056D-01
(19)	28.0000	-9.19186D-02
(20)	29.0000	-8.36685D-02
(21)	30.0000	-7.62455D-02

***** LEAST-SQUARES CHI SQUARED AT FORMER VALUES OF PARAMETERS IS 3.042660

CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS 0.169037

Table V.1.4. (Continued)

***** BAYESIAN WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL
(1)	10.0000	7.98260D-03
(2)	11.0000	4.46770D-03
(3)	12.0000	1.24752D-03
(4)	13.0000	-1.40679D-03
(5)	14.0000	-3.32483D-03
(6)	15.0000	-4.46246D-03
(7)	16.0000	-4.88783D-03
(8)	17.0000	-4.74093D-03
(9)	18.0000	-4.19167D-03
(10)	19.0000	-3.40165D-03
(11)	20.0000	-2.50467D-03
(12)	21.0000	-1.59891D-03
(13)	22.0000	-7.47717D-04
(14)	23.0000	1.27230D-05
(15)	24.0000	6.67007D-04
(16)	25.0000	1.21289D-03
(17)	26.0000	1.65624D-03
(18)	27.0000	2.00696D-03
(19)	28.0000	2.27718D-03
(20)	29.0000	2.47889D-03
(21)	30.0000	2.62336D-03

***** BAYESIAN CHI SQUARED AT FORMER VALUES OF PARAMETERS IS 0.261103

CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS 0.014506

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	1.000000	1.048524
2	-25.000000	-25.178821
3	300.000000	295.829072

***** THEORETICAL CALCULATION

	ENERGY	DATA	THEORY
1	10.000000	154.000000	153.981201
2	11.000000	151.900000	151.889852
3	12.000000	152.000000	151.997309
4	13.000000	154.299999	154.303573
5	14.000000	158.799999	158.808644
6	15.000000	165.500000	165.512522
7	16.000000	174.400000	174.415206
8	17.000000	185.500000	185.516698
9	18.000000	198.799999	198.816996
10	19.000000	214.299999	214.316101
11	20.000000	232.000000	232.014013
12	21.000000	251.900000	251.910731
13	22.000000	274.000000	274.006257
14	23.000000	298.299999	298.300589
15	24.000000	324.799999	324.793728
16	25.000000	353.500000	353.485674
17	26.000000	384.400002	384.376426
18	27.000000	417.500000	417.465986
19	28.000000	452.799999	452.754352
20	29.000000	490.299999	490.241525
21	30.000000	530.000000	529.927505

Table V.1.4. (Continued)

****** PARTIAL DERIVATIVES

	ENERGY	1	2	3
1	1.000000D+01	2.09705D+02	1.00000D+01	1.00000D+00
2	1.100000D+01	2.53743D+02	1.10000D+01	1.00000D+00
3	1.200000D+01	3.01975D+02	1.20000D+01	1.00000D+00
4	1.300000D+01	3.54401D+02	1.30000D+01	1.00000D+00
5	1.400000D+01	4.11022D+02	1.40000D+01	1.00000D+00
6	1.500000D+01	4.71836D+02	1.50000D+01	1.00000D+00
7	1.600000D+01	5.36844D+02	1.60000D+01	1.00000D+00
8	1.700000D+01	6.06047D+02	1.70000D+01	1.00000D+00
9	1.800000D+01	6.79444D+02	1.80000D+01	1.00000D+00
10	1.900000D+01	7.57035D+02	1.90000D+01	1.00000D+00
11	2.000000D+01	8.38820D+02	2.00000D+01	1.00000D+00
12	2.100000D+01	9.24799D+02	2.10000D+01	1.00000D+00
13	2.200000D+01	1.01497D+03	2.20000D+01	1.00000D+00
14	2.300000D+01	1.10934D+03	2.30000D+01	1.00000D+00
15	2.400000D+01	1.20790D+03	2.40000D+01	1.00000D+00
16	2.500000D+01	1.31066D+03	2.50000D+01	1.00000D+00
17	2.600000D+01	1.41760D+03	2.60000D+01	1.00000D+00
18	2.700000D+01	1.52875D+03	2.70000D+01	1.00000D+00
19	2.800000D+01	1.64409D+03	2.80000D+01	1.00000D+00
20	2.900000D+01	1.76362D+03	2.90000D+01	1.00000D+00
21	3.000000D+01	1.88734D+03	3.00000D+01	1.00000D+00

***** LEAST-SQUARES WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL		ENERGY	RESIDUAL
(1)	10.0000	7.92658D-03	(13)	22.0000	-8.33360D-04
(2)	11.0000	4.39804D-03	(14)	23.0000	-6.62532D-05
(3)	12.0000	1.16473D-03	(15)	24.0000	5.94473D-04
(4)	13.0000	-1.50108D-03	(16)	25.0000	1.14645D-03
(5)	14.0000	-3.42808D-03	(17)	26.0000	1.59546D-03
(6)	15.0000	-4.57159D-03	(18)	27.0000	1.95140D-03
(7)	16.0000	-4.99965D-03	(19)	28.0000	2.22638D-03
(8)	17.0000	-4.85250D-03	(20)	29.0000	2.43242D-03
(9)	18.0000	-4.30057D-03	(21)	30.0000	2.58080D-03
(10)	19.0000	-3.50607D-03			
(11)	20.0000	-2.60339D-03			
(12)	21.0000	-1.69123D-03			

***** LEAST-SQUARES CHI SQUARED AT FORMER VALUES OF PARAMETERS IS 0.001191

CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS 0.000066

Table V.1.4. (Continued)

***** BAYESIAN WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL
(1)	10.0000	7.99345D-03
(2)	11.0000	4.47432D-03
(3)	12.0000	1.25021D-03
(4)	13.0000	-1.40741D-03
(5)	14.0000	-3.32791D-03
(6)	15.0000	-4.46708D-03
(7)	16.0000	-4.89315D-03
(8)	17.0000	-4.74624D-03
(9)	18.0000	-4.19647D-03
(10)	19.0000	-3.40564D-03
(11)	20.0000	-2.50769D-03
(12)	21.0000	-1.60092D-03
(13)	22.0000	-7.48777D-04
(14)	23.0000	1.25300D-05
(15)	24.0000	6.67567D-04
(16)	25.0000	1.21409D-03
(17)	26.0000	1.65796D-03
(18)	27.0000	2.00909D-03
(19)	28.0000	2.27964D-03
(20)	29.0000	2.48159D-03
(21)	30.0000	2.62624D-03

***** BAYESIAN CHI SQUARED AT FORMER VALUES OF PARAMETERS IS 0.261097

CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS 0.014505

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	1.000000	1.048523	0.007940
2	-25.000000	-25.178796	0.607752
3	300.000000	295.828865	5.208788

	STD. DEV.	CORRELATION 1	2	3
1	7.940102D-03	100		
2	6.077518D-01	-99	100	
3	5.208788D+00	95	-99	100

Table V.1.5. Teletype Commands and User Responses to Program Prompts for Example 1, Using Automatic Numerical Derivative Option.

```
.EX BAYES,EX1,SHOW,DPPXX,DO4NML↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION?  5,.001↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES?  Y↓
WHAT IS FRACTION DIFFERENCE FOR AUTOMATIC DERIVATIVE?  .001↓
INITIAL GUESS FOR PARAMETERS?  1.,-25.,300.↓
WHAT'S NAME OF THE DATA FILE?  EX1.DAT↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.1.6. Output Resulting from Use of the Commands Shown in Table V.1.5, with Calls to Subroutines ØUTDAT, ØUTREL, ØUTCHL, ØUTREB, ØUTCHB, ØUTGMG, ØUTTH and ØUTG Suppressed to Minimize Output.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS    0.001000
AUTOMATIC DERIVATIVE USES STEP SIZE    0.00100

******INPUT PARAMETER VALUES

         PARAMETER        UNCERTAINTY
   1      1.000000        0.100000
   2    -25.000000        2.500000
   3    300.000000       30.000000

***** INTERMEDIATE RESULTS

         OLD PARAM        NEW PARAMETERS
   1      1.000000         1.049677
   2    -25.000000       -25.177087
   3    300.000000       295.814766

***** INTERMEDIATE RESULTS

         OLD PARAM        NEW PARAMETERS
   1      1.000000         1.048524
   2    -25.000000       -25.178821
   3    300.000000       295.829072

******* NEW VALUES FOR PARAMETERS

            POLD            PNEW        UNCERTAINTY
   1      1.000000         1.048523       0.007940
   2    -25.000000       -25.178796       0.607752
   3    300.000000       295.828865       5.208788

        STD. DEV.     CORRELATION

                       1   2   3
 1  7.940102D-03 100
 2  6.077518D-01 -99 100
 3  5.208788D+00  95 -99 100
```

Table V.1.7. File BAYES.ØUT, Which Suppresses Calls to the Subroutines Listed in the File.

OUTDAT	INPUT DATA IS NOT PRINTED
OUTREL	LEAST-SQUARES RESIDUALS ARE NOT PRINTED
OUTCHL	LEAST-SQUARES CHI-SQUARED IS NOT PRINTED
OUTREB	BAYESIAN RESIDUALS ARE NOT PRINTED
OUTCHB	BAYESIAN CHI-SQUARED IS NOT PRINTED
OUTTH	THEORETICAL VALUES ARE NOT PRINTED
OUTG	PARTIAL DERIVATIVES ARE NOT PRINTED
OUTGMG	MATRIX GMG IS NOT LISTED IN FILE FOR28.DAT

Example 2. Combine Independent Data Sets

The Problem:

Two or more experimental data sets contain the same or related information but were obtained from independent sources. Combine the data sets to obtain the best value for each quantity on which any information is available.

The Solution:

The model parameters P are set equal to the quantities of interest. When the information in the data set is a measurement of one specific quantity, the theoretical value T is equal to the parameter (T=P); the derivative with respect to that parameter is unity (G=1.0) and the derivative with respect to all other parameters is zero. When the information in the data set is a measurement of relationships between two or more quantitites (e.g., a relative cross section), theoretical values and derivatives are adjusted accordingly.

Specific Case:

Data set 1 consists of the total cross section for neutrons of fixed energy impinging on target element A, the cross section for element B relative to A, and uncertainties and correlations on both numbers. Data set 2 contains results of an independent measurement of the cross sections for element B, the cross section for element A relative to B, the cross section for element C relative to B, and the associated uncertainties and correlations. Find the resulting best estimate of the cross sections for

A, B, and C, and the uncertainties and correlations. (Note: the program GLUCS [HE80] was written to solve problems similar to this in evaluation of neutron cross sections for the Evaluated Nuclear Data File (ENDF) files. GLUCS uses a preliminary version of BAYES for solution of Bayes' equations.)

Let the model parameters be σ_A, σ_B, and σ_C, the cross sections for elements A, B, and C, respectively. The data are now a function of <u>two</u> independent parameters (i.e., the element types) rather than one (energy, in the previous example). The data sets might be typed in files as shown in Table V.2.1. The first two columns give the element types; where they differ, the data refer to the ratio of first to second. Where they agree, the data is an absolute cross section. Following the blank line, the data covariance matrix is given.

The theoretical value T depends on the element types involved. For example, in data set 1 the first theoretical value is $T_1 = \sigma_A$ and the second is $T_2 = \sigma_B / \sigma_A$. The partial derivative matrix is then

$$G = \begin{bmatrix} 1 & 0 & 0 \\ -\sigma_B/\sigma_A^2 & 1/\sigma_A & 0 \end{bmatrix}$$

where data points correspond to rows and parameters to columns.

The PDP-10 file EX2.F4 shown in Table V.2.2 contains the four user-supplied subroutines needed to solve this problem. Table V.2.3 shows the modified versions of ØUTDAT, ØUTTH, ØUTG, and ØUTREL, required to obtain sensible output in this case where the one independent variable (e.g., energy) is replaced by two independent variables which are, moreover, alphanumeric. This table is part of file EX2ØUT.F4, which also contains the remaining (unmodified) output subroutines.

For this example, no prior information is known for the parameter values. To approximate this situation for program BAYES, the diagonal elements of the parameter covariance matrix are initialized at $(1.0 \times P)^2$.

Commands to execute this program and responses to program prompts are shown in Table V.2.4. The output file is shown in Table V.2.5. Identical results (to six significant digits in the parameters, and three in the uncertainties) are obtained by reversing the order of data input. An equivalent approach is to generate the combined data file EX23.DAT, shown in Table V.2.6, and execute the program with this data set only. The output file from this run is shown in Table V.2.7. Note that final values obtained for the three quantities of interest (σ_A,σ_B, and σ_C, or equivalently P_1, P_2, and P_3) and their covariance matrix are identical no matter what method of data input is used.

Table V.2.1. Input Data for Example 2. Data Set 1 is Contained in PDP-10 File EX21.DAT, Data Set 2 in EX22.DAT

Data Set 1		Data Set 2		
AA	10.30	BB	11.50	
BA	1.20	AB	0.80	
		CB	1.40	
0.7				
.03	0.05	.04	.02	.01
		.02	.08	.03
		.01	.03	.08

Table V.2.2. User-Supplied Subroutines for Example 2, Found on File EX2.F4 on the ØRELA PDP-10.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARAM, PARCOV, AND IFPAR (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C       NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 3
      PARAM(1) = 10.0D0
      PARAM(2) = 12.0D0
      PARAM(3) = 17.0D0
C
      DO 20 I=1,NPARAM
           DO 10 L=1,I
                PARCOV(L,I) = 0.D0
                IF (L.EQ.I) PARCOV(L,I) = PARAM(I)**2
   10      CONTINUE
   20 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
C       NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /
```

Table V.2.2. (Continued)

```
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME)
C
      N = 0
   10 READ (24,99997,END=20) A, B, C
      IF (A.EQ.DBLANK) GO TO 20
      N = N + 1
      GO TO 10
   20 CONTINUE
      NDAT = N
      REWIND 24
C
C
      DO 30 I=1,NDAT
           READ (24,99997) A, B, C
           E(I) = A
           E2(I) = B
           DATA(I) = C
   30 CONTINUE
C
      READ (24,99997) A
      DO 40 I=1,NDAT
           READ (24,99996) (DATCOV(L,I),L=1,I)
   40 CONTINUE
      CLOSE (UNIT=24)
      RETURN
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (2A1, F)
99996 FORMAT (10F)
      END
C
C
C
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C       NML, JANUARY 1981
C
```

Table V.2.2. (Continued)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DIMENSION ELMNT(3)
      DATA ELMNT /1HA,1HB,1HC/
      DO 10 I=1,3
           IF (E(KDAT).NE.ELMNT(I)) GO TO 10
           THEO = PARAM(I)
   10 CONTINUE
      IF (E(KDAT).EQ.E2(KDAT)) GO TO 30
      DO 20 I=1,3
           IF (E2(KDAT).NE.ELMNT(I)) GO TO 20
           THEO = THEO/PARAM(I)
   20 CONTINUE
   30 CONTINUE
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR PARAMETER KPAR,
C ***            AT ENERGY E(KDAT)
C
C        NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DIMENSION ELMNT(3)
      DATA ELMNT /1HA,1HB,1HC/
C
      I=KPAR
      GG=0.
      IF (E(KDAT).NE.E2(KDAT)) GO TO 10
      IF (E(KDAT).EQ.ELMNT(I)) GG = 1.0D0
      GO TO 30
```

Table V.2.2. (Continued)

```
10 CONTINUE
   DO 20 J=1,3
        IF (E(KDAT).EQ.ELMNT(J)) I1=J
        IF (E2(KDAT).EQ.ELMNT(J)) I2=J
20 CONTINUE
   IF (I.EQ.I1) GG = 1.DO/PARAM(I2)
   IF (I.EQ.I2) GG = -PARAM(I1)/(PARAM(I2)**2)
30 CONTINUE
   DERIV = GG
   RETURN
   END
```

Table V.2.3. Modified Versions of Subroutines ØUTDAT, ØUTTH, ØUTG, and ØUTREL, for Use in Example 2.

```
C
C
C
      SUBROUTINE OUTDAT
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), DATCOV(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DIMENSION IDDUM(1)
      EQUIVALENCE(IDDUM(1),SIG(1))
C
      WRITE (6,99999)
C
      II = 0
      DO 10 I=1,NDAT
           II = II + I
           DUM(I) = DSQRT(DATCOV(II))
   10 CONTINUE
C
      WRITE (6,99998)
      WRITE (6,99997) (I,E(I),E2(I),DATA(I),DUM(I),I=1,NDAT)
C
      II = 0
      IOFF = 0
      DO 30 I=1,NDAT
           DO 20 J=1,I
                II = II + 1
                IF (I.EQ.J) GO TO 30
                IF (DATCOV(II).NE.0.D0) IOFF = IOFF + 1
   20      CONTINUE
   30 CONTINUE
      IF (IOFF.NE.0) CALL OUTV(DATCOV, DUM, IDDUM, NDAT)
      RETURN
C
99999 FORMAT (25H0****** INPUT DATA VALUES)
99998 FORMAT (48H0        DATA POINT        VALUE     UNCERTAINTY)
99997 FORMAT (I5, 5X, A5, A4, 2F14.6)
      END
C
C
C
```

Table V.2.3. (Continued)

```
      SUBROUTINE OUTTH
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), DATCOV(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      WRITE (6,99999)
      WRITE (6,99998)
      WRITE (6,99997) (I,E(I),E2(I),DATA(I),TH(I),I=1,NDAT)
      RETURN
C
99999 FORMAT (30H0***** THEORETICAL CALCULATION)
99998 FORMAT (45H0          TYPE          DATA          THEORY)
99997 FORMAT (I5, 5X, A5, A4, 2F14.6)
      END
C
C
C
      SUBROUTINE OUTG
C
C *** PURPOSE -- OUTPUT E VS G
C
C       NML, AUGUST 1980
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), DATCOV(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DATA T2 /10HDATA TYPE /
C
      WRITE (6,99999)
C
      MIN = 1
      MAX = MIN0(NPAR,7)
   10 WRITE (6,99998)
      WRITE (6,99997) T2, (I,I=MIN,MAX)
      DO 20 J=1,NDAT
          WRITE (6,99996) J, E(J), E2(J), (G(J,I),I=MIN,MAX)
   20 CONTINUE
```

Table V.2.3. (Continued)

```
C
      IF (MAX.EQ.NPAR) GO TO 30
      MIN = MAX + 1
      MAX = MAX + 7
      IF (MAX.GT.NPAR) MAX = NPAR
      GO TO 10
   30 RETURN
C
99999 FORMAT (27H0****** PARTIAL DERIVATIVES)
99998 FORMAT (20X)
99997 FORMAT (10X, A10, I10, 6I14)
99996 FORMAT (I4, 2X, 5X, A5, A4, 2X, 7G14.5)
      END
C
C
C
      SUBROUTINE OUTREL
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), EXPT(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DIMENSION FORMT(4)
      DATA PL /1H(/, PR /1H)/
      DATA FORMT /40H((1X,4(A1,I3,A1,2X,2A3,1PG13.5,2X))    /
      DATA T1 /10HDATA TYPE /, T2 /10HRESIDUAL  /
C
C
      WRITE (6,99999)
      GO TO 10
C
      ENTRY OUTREB
      WRITE (6,99998)
C
   10 WRITE (6,99997) (T1,T2,I=1,4)
C
      N = NDAT
      M = N/4
      MM = M*4
      IF (MM.NE.N) GO TO 20
      MM = M
      GO TO 30
```

Table V.2.3. (Continued)

```
   20 M = M + 1
      K = MM + 4 - N
      IF (K.GE.M) GO TO 70
      MM = M - K
   30 CONTINUE
      IM = 3*M
      DO 40 I=1,MM
           IM = IM + 1
           WRITE (6,FORMT) (PL,L,PR,E(L),E2(L),DUM(L),L=I,IM,M)
   40 CONTINUE
      IF (MM.EQ.M) GO TO 60
      IM = 2*M + MM
      MM = MM + 1
      DO 50 I=MM,M
           IM = IM + 1
           WRITE (6,FORMT) (PL,L,PR,E(L),E2(L),DUM(L),L=I,IM,M)
   50 CONTINUE
   60 CONTINUE
      RETURN
   70 WRITE (6,FORMT) (PL,I,PR,E(I),E2(I),DUM(I),I=1,N)
      RETURN
99999 FORMAT (39H0***** LEAST-SQUARES WEIGHTED RESIDUALS,
     *     31H AT FORMER VALUES OF PARAMETERS)
99998 FORMAT (34H0***** BAYESIAN WEIGHTED RESIDUALS, 8H AT FORM,
     *     23HER VALUES OF PARAMETERS)
99997 FORMAT (3H0  , 4(2X, A10, 2X, A10, 4X))
      END
```

Table V.2.4. Teletype Commands and User Responses to Program Prompts for Example 2.

```
.EX BAYES,EX2,EX2OUT,DPPXX,DO4NML ↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001 ↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N ↓
WHAT'S NAME OF THE DATA FILE? EX21.DAT ↓
WHAT'S NAME OF THE DATA FILE? EX22.DAT ↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.2.5. Output Resulting from the Use of Commands Given in Table V.2.4.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

******INPUT PARAMETER VALUES

        PARAMETER      UNCERTAINTY
   1    10.000000      10.000000
   2    12.000000      12.000000
   3    17.000000      17.000000

****** INPUT DATA VALUES

       DATA POINT          VALUE      UNCERTAINTY
   1     A    A         10.300000      0.836660
   2     B    A          1.200000      0.223607

      STD. DEV.     CORRELATION

                     1   2
  1  8.366600D-01  100
  2  2.236068D-01   16  100

***** THEORETICAL CALCULATION

        TYPE             DATA          THEORY
   1     A    A       10.300000      10.000000
   2     B    A        1.200000       1.200000

****** PARTIAL DERIVATIVES

        DATA TYPE           1              2              3
  1       A    A       0.10000D+01    0.00000D+00    0.00000D+00
  2       B    A      -0.12000D+00    0.10000D+00    0.00000D+00

***** INTERMEDIATE RESULTS

       OLD PARAM       NEW PARAMETERS
   1    10.000000      10.295236
   2    12.000000      12.340715
   3    17.000000      17.000000

***** THEORETICAL CALCULATION

        TYPE             DATA          THEORY
   1     A    A       10.300000      10.295236
   2     B    A        1.200000       1.198682
```

Table V.2.5. (Continued)

****** PARTIAL DERIVATIVES

	DATA	TYPE	1	2	3
1	A	A	0.10000D+01	0.00000D+00	0.00000D+00
2	B	A	-0.11643D+00	0.97132D-01	0.00000D+00

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	10.000000	10.295223	0.828571
2	12.000000	12.339970	2.589452
3	17.000000	17.000000	17.000000

	STD. DEV.	CORRELATION 1	2	3
1	8.285711D-01	100		
2	2.589452D+00	51	100	
3	1.700000D+01	0	0	100

****** INPUT DATA VALUES

	DATA	POINT	VALUE	UNCERTAINTY
1	B	B	11.500000	0.200000
2	A	B	0.800000	0.282843
3	C	B	1.400000	0.282843

	STD. DEV.	CORRELATION 1	2	3
1	2.000000D-01	100		
2	2.828427D-01	35	100	
3	2.828427D-01	18	38	100

***** THEORETICAL CALCULATION

	TYPE		DATA	THEORY
1	B	B	11.500000	12.339970
2	A	B	0.800000	0.834299
3	C	B	1.400000	1.377637

****** PARTIAL DERIVATIVES

	DATA	TYPE	1	2	3
1	B	B	0.00000D+00	0.10000D+01	0.00000D+00
2	A	B	0.81037D-01	-0.67609D-01	0.00000D+00
3	C	B	0.00000D+00	-0.11164D+00	0.81037D-01

Table V.2.5. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	10.295223	10.126092
2	12.339970	11.525014
3	17.000000	16.526001

***** THEORETICAL CALCULATION

	TYPE		DATA	THEORY
1	B	B	11.500000	11.525014
2	A	B	0.800000	0.878619
3	C	B	1.400000	1.433925

****** PARTIAL DERIVATIVES

	DATA	TYPE	1	2	3
1	B	B	0.00000D+00	0.10000D+01	0.00000D+00
2	A	B	0.86768D-01	-0.76236D-01	0.00000D+00
3	C	B	0.00000D+00	-0.12442D+00	0.86768D-01

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	10.295223	10.122322
2	12.339970	11.525954
3	17.000000	16.495590

***** THEORETICAL CALCULATION

	TYPE		DATA	THEORY
1	B	B	11.500000	11.525954
2	A	B	0.800000	0.878220
3	C	B	1.400000	1.431169

****** PARTIAL DERIVATIVES

	DATA	TYPE	1	2	3
1	B	B	0.00000D+00	0.10000D+01	0.00000D+00
2	A	B	0.86761D-01	-0.76195D-01	0.00000D+00
3	C	B	0.00000D+00	-0.12417D+00	0.86761D-01

Table V.2.5. (Continued)

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	10.295223	10.122325	0.701516
2	12.339970	11.525953	0.184291
3	17.000000	16.495587	3.005402

	STD. DEV.	CORRELATION 1	2	3
1	7.015164D-01	100		
2	1.842914D-01	13	100	
3	3.005402D+00	10	13	100

Table V.2.6. Data and Covariance Matrix for the Combination of the Two Data Sets Shown in Table V.2.1.

AA	10.3
BA	1.2
BB	11.50
AB	0.80
CB	1.40

0.7	0.03	0.0	0.0	0.0
0.03	0.05	0.0	0.0	0.0
0.0	0.0	0.04	0.02	0.01
0.0	0.0	0.02	0.08	0.03
0.0	0.0	0.01	0.03	0.08

Table V.2.7. Output Obtained by Including All Five Data Points Simultaneously.

MAXIMUM NUMBER OF ITERATIONS IS 5
CONVERGENCE FACTOR IS 0.001000

******INPUT PARAMETER VALUES

	PARAMETER	UNCERTAINTY
1	10.000000	10.000000
2	12.000000	12.000000
3	17.000000	17.000000

****** INPUT DATA VALUES

	DATA	POINT	VALUE	UNCERTAINTY
1	A	A	10.300000	0.836660
2	B	A	1.200000	0.223607
3	B	B	11.500000	0.200000
4	A	B	0.800000	0.282843
5	C	B	1.400000	0.282843

	STD. DEV.	CORRELATION 1	2	3	4	5
1	8.366600D-01	100				
2	2.236068D-01	16	100			
3	2.000000D-01	0	0	100		
4	2.828427D-01	0	0	35	100	
5	2.828427D-01	0	0	18	38	100

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	10.000000	10.119930
2	12.000000	11.525405
3	17.000000	16.493671

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	10.000000	10.122885	0.705865
2	12.000000	11.526101	0.184283
3	17.000000	16.496059	3.005421

	STD. DEV.	CORRELATION 1	2	3
1	7.058654D-01	100		
2	1.842831D-01	13	100	
3	3.005421D+00	10	13	100

Example 3. Background Subtraction

The Problem:

Raw data points X_i are independent, with uncertainties ΔX_i. However, there is a known background B with uncertainty ΔB which must be subtracted from the raw data. This background is independent of the data.

The First Solution:

There are at least two ways to properly treat the background subtraction. The first is to generate new "data" D_i by subtracting the background explicitly, and find the covariance matrix V_{ij} for this data. That is, we may set

$$D_i = X_i - B \qquad (V.3.1)$$

so that changes in either X_i or B lead to changes in D_i:

$$\delta D_i = \delta X_i - \delta B. \qquad (V.3.2)$$

The variance for D_i is found by taking the expectation value of the square of this error, i.e.,

$$V_{ii} = \langle(\delta D)^2\rangle = \langle(\delta X_i - \delta B)^2\rangle = \Delta X_i^{\,2} + \Delta B^2 \qquad (V.3.3)$$

since X_i and B are assumed independent. The covariance between points i and j can similarly be found as

$$V_{ij} = \langle(\delta X_i - \delta B)(\delta X_j - \delta B)\rangle. \qquad (V.3.4)$$

For $i \neq j$, only the term $< \delta B \delta B >$ remains. This term is the covariance of the background alone; thus the covariance V_{ij} reduces to

$$V_{ij} = \Delta B^2. \tag{V.3.5}$$

The Second Solution:

An alternative approach to the problem of background subtraction is to treat raw data points X_i as the data to be fit, and modify the theoretical calculations appropriately. That is, where in the first approach we generated theoretical values T, here we generate

$$T' = T + b \tag{V.3.6}$$

where b is a parameter of our model, and is input as the known value B with uncertainty ΔB. The derivative G associated with parameter b is unity.

An important advantage of this approach is that the data points remain uncorrelated. Thus a large data set can be broken into several smaller ones (to conform to computer limitations) and analyzed sequentially. Results will be equivalent to those which would be obtained if the entire set could be analyzed simultaneously. This is illustrated by the sample problem shown below, where we perform the calculation both simultaneously and sequentially.

Specific Case:

Artificially generated "raw data" are shown in Table V.3.1 (the PDP-10 file EX31.DAT). The background level was determined from an independent "measurement" to be approximately $B = 40.17 \pm 6.77$.

For the first method of solution, we wish to fit the adjusted data with a function of the form

$$f(E,P) = P_1 \exp \{-(E-P_2)^2/P_3^2\}, \qquad (V.3.7)$$

remembering that the data have the covariance matrix given by Eqs. (V.3.3) and (V.3.5). User-supplied subroutines to accomplish this are stored in PDP-10 file EX31.F4, shown in Table V.3.2. Note that the raw data are read explicitly, and the background-subtracted data and adjusted data covariance matrix are generated within subroutine SETDAT.

Commands to execute this program and responses to program prompts are given in Table V.3.3, and the output file is listed in Table V.3.4. Note that the presence of off-diagonal elements of the data covariance matrix requires all data to be included simultaneously.

In the second method of solution, the functional form we wish to fit is

$$f(E,P) = P_1 \exp \{-(E-P_2)^2/P_3^2\} + P_4 \qquad (V.3.8)$$

where the fourth parameter is input as the background level 40.166463 ± 6.766185. User-supplied subroutines for this case are stored in PDP-10 file EX32.F4, shown in Table V.3.5. Execution commands are given in Table V.3.6, and output in Table V.3.7. Results for the first three parameters are exactly equivalent to those obtained via the first method; that this is to be expected is illustrated in Appendix A, Section 4, where it is demonstrated that adding a constant covariance matrix like that shown in Eqs. (V.3.4) and (V.3.5) is equivalent to adding a constant value to the theory as in Eq. (V.3.8).

With the second method it is possible to analyze the data sequentially rather than simultaneously. That is, we can analyze the data from E=25 to E=49, e.g., and use the results of that analysis as input to the analysis of the remaining data. Execution commands for this sequential analysis are shown in Table V.3.8, and results are given in Table V.3.9.

Results obtained from such a sequential analysis are not expected to be exactly the same as those obtained from a simultaneous analysis, because the approximations built into the model are not exactly valid. Nevertheless, reasonable agreement between sequential and simultaneous analysis is usually obtained. In this example, the discrepancy between the two analyses, for each of the four parameters, was less than 0.6% absolute, or (more significantly) less than 7.0% of the corresponding uncertainty. Differences in the correlation matrix are more pronounced, though trends are given consistently.

For practical situations in which the large size of the data set prohibits simultaneous analysis, changing the order of the data subsets in a sequential analysis can be used for a consistency check. In this example, analyzing the high-energy region first and then the low-energy region gave results consistent with those quoted here.

Table V.3.1. ORELA PDP-10 File EX31.DAT, Containing "Raw" Data Required for Example 3.

25.000000	30.032617	5.480202
26.000000	39.293188	6.268428
27.000000	37.558659	6.128512
28.000000	62.021808	7.875393
29.000000	33.491881	5.787217
30.000000	35.314806	5.942626
31.000000	59.671737	7.724748
32.000000	47.780402	6.912337
33.000000	55.474042	7.448090
34.000000	50.109183	7.078784
35.000000	57.132362	7.558595
36.000000	51.935085	7.206600
37.000000	67.476355	8.214399
38.000000	64.423848	8.026447
39.000000	69.447257	8.333502
40.000000	79.158297	8.897095
41.000000	86.357061	9.292850
42.000000	94.570995	9.724762
43.000000	77.289822	8.791463
44.000000	101.358525	10.067697
45.000000	82.501708	9.083045
46.000000	108.200675	10.401955
47.000000	114.508126	10.700847
48.000000	107.876942	10.386382
49.000000	130.520432	11.424554
50.000000	114.470710	10.699099
51.000000	112.498927	10.606551
52.000000	143.359819	11.973296
53.000000	130.125650	11.407263
54.000000	117.260513	10.828689
55.000000	112.144705	10.589840
56.000000	110.041392	10.490062
57.000000	111.031748	10.537160
58.000000	90.807915	9.529319
59.000000	91.758967	9.579090
60.000000	70.201249	8.378619
61.000000	78.406860	8.854765
62.000000	82.244825	9.068893
63.000000	83.963837	9.163178
64.000000	85.339459	9.237936
65.000000	67.236270	8.199773

Table V.3.1. (Continued)

66.000000	63.828962	7.989303
67.000000	64.729895	8.045489
68.000000	64.914184	8.056934
69.000000	57.713948	7.596970
70.000000	51.214544	7.156434
71.000000	36.063748	6.005310
72.000000	47.804664	6.914092
73.000000	41.833538	6.467885
74.000000	39.249423	6.264936
75.000000	31.293112	5.594025

Table V.3.2. User-Supplied Subroutines for Example 3, First Method of Solution. These routines are stored on file EX31.F4 on the ORELA PDP-10.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARAM, PARCOV, AND IFPAR (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C        NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 3
      PARAM(1) = 80.
      PARAM(2) = 50.
      PARAM(3) = 10.
      DO 20 I=1,NPARAM
           DO 10 L=1,I
                PARCOV(L,I) = 0.D0
                IF (L.EQ.I) PARCOV(L,I) = PARAM(I)**2
   10      CONTINUE
   20 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
C        NML, MARCH 1981
C
```

Table V.3.2. (Continued)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME)
C
C *** READ FILE 24 TO COUNT THE NUMBER OF DATA POINTS (NDAT)
      N = 0
   10 READ (24,99997,END=20) EE, D, ERR
      IF (EE.LE.0.D0.AND.D.LE.D0) GO TO 20
      N = N + 1
      GO TO 10
   20 CONTINUE
      NDAT = N
      REWIND 24
C
C
C *** INITIALIZE DATA COVARIANCE MATRIX DATCOV
      BACK = 40.166463D0
      DELBAK = 6.766185D0
      DELBK2 = DELBAK**2
      DO 40 I=1,NDAT
           DO 30 L=1,I
                DATCOV(L,I) = DELBK2
   30      CONTINUE
   40 CONTINUE
C
      DO 50 I=1,NDAT
           READ (24,99997) EE, D, ERR
           E(I) = EE
           DATA(I) = D - BACK
           DATCOV(I,I) = ERR**2 + DATCOV(I,I)
   50 CONTINUE
      CLOSE (UNIT=24)
      RETURN
```

Table V.3.2. (Continued)

```
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (3F12.1)
      END
C
C
C
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      EEE = E(KDAT)
      THEO = PARAM(1)*DEXP(-(EEE-PARAM(2))**2/(PARAM(3)**2))
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR PARAMETER KPAR,
C ***            AT ENERGY E(KDAT)
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      DELE = E(KDAT) - PARAM(2)
      EXPONE = DELE*DELE/(PARAM(3)*PARAM(3))
      G1 = DEXP(-EXPONE)
      THEORY = PARAM(1)*G1
      I = KPAR
      GO TO (10, 20, 30), I
```

Table V.3.2. (Continued)

```
10 GG = G1
   GO TO 40
20 GG = 2.DO*THEORY*DELE/(PARAM(3)**2)
   GO TO 40
30 GG = 2.DO*THEORY*EXPONE/PARAM(3)
40 CONTINUE
   DERIV = GG
   RETURN
   END
```

Table V.3.3. Teletype Commands and User Responses to Program Prompts for Example 3, First Solution.

```
.EX BAYES,EX31,SHOW,DPPXX,DO4NML↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N↓
WHAT'S NAME OF THE DATA FILE? EX31.DAT↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.3.4. Output Resulting From Use of Commands Shown in Table V.3.3.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS    0.001000

******INPUT PARAMETER VALUES

          PARAMETER       UNCERTAINTY
    1     80.000000      80.000000
    2     50.000000      50.000000
    3     10.000000      10.000000

***** INTERMEDIATE RESULTS

         OLD PARAM       NEW PARAMETERS
    1     80.000000      76.833487
    2     50.000000      51.298474
    3     10.000000      12.660616

***** INTERMEDIATE RESULTS

         OLD PARAM       NEW PARAMETERS
    1     80.000000      82.365721
    2     50.000000      51.495437
    3     10.000000      13.791187

***** INTERMEDIATE RESULTS

         OLD PARAM       NEW PARAMETERS
    1     80.000000      83.008461
    2     50.000000      51.481651
    3     10.000000      13.884043

******* NEW VALUES FOR PARAMETERS

            POLD           PNEW       UNCERTAINTY
    1     80.000000      83.017172       3.578796
    2     50.000000      51.481154       0.346963
    3     10.000000      13.889437       0.855672

       STD. DEV.     CORRELATION

                      1    2    3
  1  3.578796D+00  100
  2  3.469629D-01   -2  100
  3  8.556722D-01   18   -1  100
```

Table V.3.5: User-Supplied Subroutines for Example 3, Second Method for Solution. These routines are stored on file EX32.F4 on the ORELA PDP-10.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARAM, PARCOV, AND IFPAR (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C        NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 4
      PARAM(1) = 80.
      PARAM(2) = 50.
      PARAM(3) = 10.
      PARAM(4) = 40.166463
      DO 20 I=1,NPARAM
         DO 10 L=1,I
            PARCOV(L,I) = 0.D0
            IF (L.EQ.I) PARCOV(L,I) = PARAM(I)**2
   10    CONTINUE
   20 CONTINUE
      PARCOV(NPARAM,NPARAM) = 6.766185D0**2
      RETURN
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
C        NML, JANUARY 1981
C
```

Table V.3.5. (Continued)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME)
C
C *** READ FILE 24 TO COUNT THE NUMBER OF DATA POINTS (NDAT)
      N = 0
   10 READ (24,99997,END=20) EE, D, ERR
      IF (EE.LE.0.D0.AND.D.LE.D0) GO TO 20
      N = N + 1
      GO TO 10
   20 CONTINUE
      NDAT = N
      REWIND 24
C
C
C *** INITIALIZE DATA COVARIANCE MATRIX DATCOV
      DO 40 I=1,NDAT
           DO 30 L=1,I
                DATCOV(L,I) = 0.D0
   30      CONTINUE
   40 CONTINUE
C
      DO 50 I=1,NDAT
           READ (24,99997) EE, D, ERR
           E(I) = EE
           DATA(I) = D
           DATCOV(I,I) = ERR**2 + DATCOV(I,I)
   50 CONTINUE
      CLOSE (UNIT=24)
      RETURN
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (3F12.1)
      END
C
C
C
```

Table V.3.5. (Continued)

```
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C        NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      EEE = E(KDAT)
      THEO = PARAM(1)*DEXP(-(EEE-PARAM(2))**2/(PARAM(3)**2)) +
     *     PARAM(4)
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR PARAMETER KPAR,
C ***            AT ENERGY E(KDAT)
C
C        NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      DELE = E(KDAT) - PARAM(2)
      EXPONE = DELE*DELE/(PARAM(3)*PARAM(3))
      G1 = DEXP(-EXPONE)
      THEORY = PARAM(1)*G1
      I = KPAR
      GO TO (10, 20, 30, 40), I
   10 GG = G1
      GO TO 50
   20 GG = 2.D0*THEORY*DELE/(PARAM(3)**2)
      GO TO 50
   30 GG = 2.D0*THEORY*EXPONE/PARAM(3)
      GO TO 50
   40 GG = 1.D0
   50 CONTINUE
      DERIV = GG
      RETURN
      END
```

Table V.3.6. Teletype Commands and User Responses to Program Prompts for Example 3, Second Solution.

```
.EX BAYES,EX32,SHOW,DPPXX,DO4NML ↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N↓
WHAT'S NAME OF THE DATA FILE? EX31.DAT↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.3.7. Output Resulting From the Use of Commands Shown in Table V.3.6.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

******INPUT PARAMETER VALUES

          PARAMETER       UNCERTAINTY
   1      80.000000      80.000000
   2      50.000000      50.000000
   3      10.000000      10.000000
   4      40.166463       6.766185

***** INTERMEDIATE RESULTS

          OLD PARAM       NEW PARAMETERS
   1      80.000000      76.833487
   2      50.000000      51.298474
   3      10.000000      12.660616
   4      40.166463      39.996141

***** INTERMEDIATE RESULTS

          OLD PARAM       NEW PARAMETERS
   1      80.000000      82.365721
   2      50.000000      51.495437
   3      10.000000      13.791187
   4      40.166463      35.366567

***** INTERMEDIATE RESULTS

          OLD PARAM       NEW PARAMETERS
   1      80.000000      83.008461
   2      50.000000      51.481651
   3      10.000000      13.884043
   4      40.166463      34.681893

******* NEW VALUES FOR PARAMETERS

            POLD           PNEW        UNCERTAINTY
   1      80.000000      83.017172       3.578796
   2      50.000000      51.481154       0.346963
   3      10.000000      13.889437       0.855672
   4      40.166463      34.662382       3.004104
```

Table V.3.7. (Continued)

	STD. DEV.	CORRELATION 1	2	3	4
1	3.578796D+00	100			
2	3.469629D-01	-2	100		
3	8.556722D-01	18	-1	100	
4	3.004104D+00	-54	3	-84	100

Table V.3.8. Teletype Commands and User Responses to Program Prompt for Example 3, Second Solution, with Data Supplied Half at a Time.

```
.EX BAYES,EX32,SHOW,DPPXX,DO4NML↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N↓
WHAT'S NAME OF THE DATA FILE? EX31A.DAT↓
WHAT'S NAME OF THE DATA FILE? EX31B.DAT↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.3.9. Output Resulting From the Use of Commands Shown in Table V.3.8.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

******INPUT PARAMETER VALUES

          PARAMETER      UNCERTAINTY
   1     80.000000      80.000000
   2     50.000000      50.000000
   3     10.000000      10.000000
   4     40.166463       6.766185

***** INTERMEDIATE RESULTS

          OLD PARAM      NEW PARAMETERS
   1      80.000000     95.888197
   2      50.000000     55.419689
   3      10.000000     16.079597
   4      40.166463     36.777262

***** INTERMEDIATE RESULTS

          OLD PARAM      NEW PARAMETERS
   1      80.000000     94.066483
   2      50.000000     54.597251
   3      10.000000     15.730421
   4      40.166463     35.171321

***** INTERMEDIATE RESULTS

          OLD PARAM      NEW PARAMETERS
   1      80.000000     95.343956
   2      50.000000     54.879020
   3      10.000000     15.930085
   4      40.166463     35.033138

***** INTERMEDIATE RESULTS

          OLD PARAM      NEW PARAMETERS
   1      80.000000     94.718520
   2      50.000000     54.732557
   3      10.000000     15.818054
   4      40.166463     35.113582
```

Table V.3.9. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	80.000000	95.093282
2	50.000000	54.819004
3	10.000000	15.883230
4	40.166463	35.067593

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	80.000000	94.879316	29.991130
2	50.000000	54.769427	6.467141
3	10.000000	15.845681	4.555806
4	40.166463	35.094206	4.367353

STD. DEV. CORRELATION

	STD. DEV.	1	2	3	4
1	2.999113D+01	100			
2	6.467141D+00	97	100		
3	4.555806D+00	91	96	100	
4	4.367353D+00	-52	-53	-71	100

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	94.879316	79.166133
2	54.769427	51.835727
3	15.845681	14.900612
4	35.094206	33.941739

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	94.879316	82.362750
2	54.769427	51.457965
3	15.845681	13.748462
4	35.094206	35.129103

Table V.3.9. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	94.879316	82.785880
2	54.769427	51.494527
3	15.845681	13.830654
4	35.094206	34.862361

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	94.879316	82.786117	3.733574
2	54.769427	51.494671	0.347197
3	15.845681	13.830334	0.876746
4	35.094206	34.862375	3.025873

	STD. DEV.	CORRELATION			
		1	2	3	4
1	3.733574D+00	100			
2	3.471969D-01	-7	100		
3	8.767457D-01	10	4	100	
4	3.025873D+00	-48	1	-84	100

Example 4. Normalization

The Problem:

Data points X_i, with uncertainties ΔX_i, are correct up to an overall normalization N with uncertainty ΔN. This normalization is independent of energy, and is measured independently from the experiment which produced the data points X_i.

The First Solution:

Again there are two ways to treat this problem. The first approach is to generate data D_i as

$$D_i = NX_i \tag{V.4.1}$$

which gives changes in D in terms of changes in N or X_i as

$$\delta D_i = X_i \ \delta N + N \ \delta X_i . \tag{V.4.2}$$

Then the variance on D_i is given by

$$\begin{aligned} V_{ii} &= \langle (\delta D_i)^2 \rangle \\ &= X_i^2 \langle (\delta N)^2 \rangle + N^2 \langle (\delta X_i)^2 \rangle \\ &= X_i^2 \, \Delta N^2 + N^2 \, \Delta X_i^2 \end{aligned} \tag{V.4.3}$$

and the covariance between points i and j by

$$V_{ij} = \langle \delta D_i \ \delta D_j \rangle = X_i X_j \Delta N^2 \tag{V.4.4}$$

since the data points X_i are independent.

The Second Solution:

The alternative approach is to keep X_i as the data, and incorporate the normalization into the theory. That is, if T is the theoretical value calculated in the first approach, then here we have

$$T' = T/n \qquad \text{(V.4.5)}$$

where n is a model parameter which is initialized at the known value N with uncertainty ΔN. The partial derivative G of T' with respect to n is

$$G = - T/n^2 = - T'/n, \qquad \text{(V.4.6)}$$

and the partial derivatives with respect to the other parameters must be modified to give the derivative of T' rather than of T.

As in example 3, the second approach has the important advantage that correlations are not introduced into the data themselves. Thus a large data set can be broken into several smaller ones (to conform to computer limitations) and analyzed sequentially. Results will be equivalent to those which would be obtained if the entire set could be analyzed sequentially. This is illustrated by the sample problem shown below, where we perform the calculation both simultaneously and sequentially.

Specific Case:

The artificially generated "raw data" used in this example are shown in Table V.4.1 (the PDP-10 file EX41.DAT).

For the first method of solution, we wish to first adjust the data by a multiplicative constant whose value has been predetermined to be 1.14 ± 0.10, and then fit the adjusted data with a Breit-Wigner function of the form

$$f(E,P) = \frac{\pi}{k^2} \frac{P_2^2}{(E-P_1)^2 + (P_2/2.0)^2} \qquad (V.4.7)$$

where k is the projectile momentum in the center of mass system. User-supplied subroutines to accomplish this are stored in the PDP-10 file EX41.F4, shown in Table V.4.2. Note that the raw data are read explicitly, and the adjusted data (see Eq. (V.4.1)) and data covariance matrix (see Eqs. (V.4.3) and (V.4.4)) are generated within subroutine SETDAT.

Commands to execute this program and responses to program prompts are given in Table V.4.3, and the output file is listed in Table V.4.4. Note that the presence of off-diagonal elements of the data covariance matrix require all data to be included simultaneously.

In the second method of solution, the raw data is fit directly to a function of the form

$$f(E,P) = \frac{1}{P_3} \frac{\pi}{k^2} \frac{P_2^2}{(E-P_1)^2 + (P_2/2.0)^2} \qquad (V.4.8)$$

where the third parameter is input as the predetermined multiplicative constant 1.14 ± 0.10. User-supplied subroutines for this case are stored in ORELA PDP-10 file EX42.F4, shown in Table V.4.5. Execution commands are given in Table V.4.6, and output in Table V.4.7.

For both solution methods, prior values of P_1 and P_2 are set at 120. ± 120. and 30. ± 30., respectively, and the two parameters are not known to be correlated.

Results for the first and second parameters agree with those obtained via the first method of solution to within three or more significant digits, or to within 25% of the uncertainties. Exact agreement is not expected, since the linear approximation is not strictly correct for multiplicative constants.

With the second method of solution it is possible to analyze the data sequentially rather than simultaneously. The data are divided into two subsets, file EX41A.DAT containing data from E=100 to E=125 and file EX41B.DAT containing the rest. Commands to analyze these subsets sequentially are shown in Table V.4.8, and the results given in Table V.4.9.

Agreement between parameter values for simultaneous vs sequential solutions is to within 3% of the corresponding uncertainty. This is better agreement than that obtained between the first and second solution methods (compare Tables V.4.4 and V.4.7). The correlation matrices consistently show that parameter pairs 1-2 and 1-3 are essentially uncorrelated, while the pair 2-3 are highly correlated.

For practical situations in which the large size of the data set prohibits simultaneous analysis, changing the order of the data sets in a sequential analysis can be used for a consistency check. In this example, analyzing the high-energy region first and then the low-energy region gives results similar with those quoted here, though the discrepancy between simultaneous and sequential results ranges as high as 13% of the uncertainty.

Table V.4.1. ORELA PDP-10 File EX41.DAT, Containing "Raw" Data Used for Example 4.

Energy	Data	Absolute Uncertainty
100.000000	4.223518	0.205512
101.000000	4.466085	0.211331
102.000000	4.915435	0.221708
103.000000	5.394355	0.232258
104.000000	5.503246	0.234590
105.000000	6.116313	0.247312
106.000000	6.825470	0.261256
107.000000	7.921634	0.281454
108.000000	7.883934	0.280783
109.000000	7.861386	0.280382
110.000000	9.047163	0.300785
111.000000	9.291409	0.304818
112.000000	10.681179	0.326821
113.000000	12.041003	0.347001
114.000000	12.339183	0.351272
115.000000	13.292924	0.364595
116.000000	14.269995	0.377756
117.000000	15.627672	0.395319
118.000000	16.706080	0.408731
119.000000	16.959656	0.411821
120.000000	18.299793	0.427783
121.000000	18.533516	0.430506
122.000000	18.902473	0.434770
123.000000	18.480649	0.429891
124.000000	18.769973	0.433243
125.000000	18.380886	0.428729
126.000000	16.777637	0.409605
127.000000	16.465754	0.405780
128.000000	15.647785	0.395573
129.000000	14.189659	0.376692
130.000000	13.176671	0.362997
131.000000	12.202733	0.349324
132.000000	11.513750	0.339319
133.000000	10.142552	0.318474
134.000000	9.512503	0.308423
135.000000	8.488945	0.291358
136.000000	7.840551	0.280010
137.000000	6.427479	0.253525
138.000000	6.133853	0.247666
139.000000	5.827105	0.241394

Table V.4.1. (Continued)

Energy	Data	Absolute Uncertainty
140.000000	5.353455	0.231375
141.000000	4.331043	0.208112
142.000000	4.382364	0.209341
143.000000	4.048990	0.201221
144.000000	3.694872	0.192220
145.000000	3.480069	0.186549
146.000000	3.182577	0.178398
147.000000	2.816217	0.167816
148.000000	2.787700	0.166964
149.000000	2.706788	0.164523
150.000000	2.402707	0.155007

Table V.4.2. User-Supplied Subroutines for the First Method of Solution for Example 4. These routines are stored on PDP-10 file EX41.F4.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARAM, PARCOV, AND IFPAR (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 2
      PARAM(1) = 120.
      PARAM(2) = 30.
      DO 20 I=1,NPARAM
           DO 10 L=1,I
                PARCOV(L,I) = 0.D0
                IF (L.EQ.I) PARCOV(L,I) = PARAM(I)**2
   10      CONTINUE
   20 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /
```

Table V.4.2. (Continued)

```
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME)
C
C *** READ FILE 24 TO COUNT THE NUMBER OF DATA POINTS (NDAT)
      N = 0
   10 READ (24,99997,END=20) EE, D, ERR
      IF (EE.LE.0.D0.AND.D.LE.D0) GO TO 20
      N = N + 1
      GO TO 10
   20 CONTINUE
      NDAT = N
      REWIND 24
C
C
      ANRM = 1.14D0
      ANRM2 = ANRM**2
      DELNRM = 0.10D0
      DELNR2 = DELNRM**2
C
      DO 30 I=1,NDAT
           READ (24,99997) EE, D, ERR
           E(I) = EE
           DATA(I) = D
           E2(I) = ERR
   30 CONTINUE
      CLOSE (UNIT=24)
C
      DO 50 I=1,NDAT
           DO 40 L=1,I
                DATCOV(L,I) = DATA(I)*DATA(L)*DELNR2
                IF (L.EQ.I) DATCOV(L,I) = DATCOV(L,I) +
     *               ANRM2*E2(I)**2
   40      CONTINUE
   50 CONTINUE
C
      DO 60 I=1,NDAT
           DATA(I) = DATA(I)*ANRM
   60 CONTINUE
C
      RETURN
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (3F12.1)
      END
C
C
C
```

Table V.4.2. (Continued)

```
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DATA IN /0/
      IF (IN.GT.0) GO TO 10
      AM = 100.
      CAY = 2.1968E-4*(AM/(AM+1.009))
      CAYS = 0.0314159/CAY**2/1000.
      IN = 1
   10 EEE = E(KDAT)
      THEO = (CAYS/EEE)*PARAM(2)**2/((EEE-PARAM(1))**2+0.25*
     *     PARAM(2)**2)
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR PARAMETER KPAR,
C ***            AT ENERGY E(KDAT)
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DATA IN /0/
      IF (IN.GT.0) GO TO 10
      AM = 100.
      CAY = 2.1968E-4*(AM/(AM+1.009))
      CAYS = 0.0314159/CAY**2/1000.
      IN = 1
C
```

Table V.4.2. (Continued)

```
 10 EEE = E(KDAT)
    DELE = EEE - PARAM(1)
    D = DELE**2 + 0.25*PARAM(2)**2
    B = (CAYS/EEE)*PARAM(2)/D
    D = B*DELE/D
    I = KPAR
    GO TO (20, 30), I
 20 GG = 2.*D*PARAM(2)
    GO TO 40
 30 GG = 2.0*D*DELE
 40 CONTINUE
    DERIV = GG
    RETURN
    END
```

Table V.4.3. Teletype Commands and User Responses to Program Prompts for Example 4, First Method of Solution.

```
.EX BAYES,EX41,SHOW,DPPXX,DO4NML ↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001 ↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N ↓
WHAT'S NAME OF THE DATA FILE? EX41.DAT ↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.4.4. Output Resulting from the Use of Commands Given in Table V.4.3.

MAXIMUM NUMBER OF ITERATIONS IS 5
CONVERGENCE FACTOR IS 0.001000

******INPUT PARAMETER VALUES

	PARAMETER	UNCERTAINTY
1	120.000000	120.000000
2	30.000000	30.000000

****** INPUT DATA VALUES

	DATA POINT	VALUE	UNCERTAINTY
1	100.000000	4.814811	0.482980
2	101.000000	5.091337	0.507445
3	102.000000	5.603596	0.552717
4	103.000000	6.149565	0.600913
5	104.000000	6.273700	0.611864
6	105.000000	6.972597	0.673484
7	106.000000	7.781036	0.744697
8	107.000000	9.030663	0.854677
9	108.000000	8.987685	0.850896
10	109.000000	8.961980	0.848635
11	110.000000	10.313766	0.967517
12	111.000000	10.592206	0.991995
13	112.000000	12.176544	1.131233
14	113.000000	13.726743	1.267415
15	114.000000	14.066669	1.297272
16	115.000000	15.153933	1.392758
17	116.000000	16.267794	1.490564
18	117.000000	17.815546	1.626450
19	118.000000	19.044931	1.734371
20	119.000000	19.334008	1.759746
21	120.000000	20.861764	1.893845
22	121.000000	21.128208	1.917231
23	122.000000	21.548819	1.954147
24	123.000000	21.067940	1.911941
25	124.000000	21.397769	1.940890
26	125.000000	20.954210	1.901959
27	126.000000	19.126506	1.741532
28	127.000000	18.770960	1.710321
29	128.000000	17.838475	1.628463
30	129.000000	16.176211	1.482523

Table V.4.4. (Continued)

31	130.000000	15.021405	1.381119
32	131.000000	13.911116	1.283610
33	132.000000	13.125675	1.214618
34	133.000000	11.562509	1.077277
35	134.000000	10.844253	1.014151
36	135.000000	9.677397	0.911561
37	136.000000	8.938228	0.846545
38	137.000000	7.327326	0.704739
39	138.000000	6.992592	0.675246
40	139.000000	6.642900	0.644423
41	140.000000	6.102939	0.596798
42	141.000000	4.937389	0.493828
43	142.000000	4.995895	0.499003
44	143.000000	4.615849	0.465364
45	144.000000	4.212154	0.429580
46	145.000000	3.967279	0.407843
47	146.000000	3.628138	0.377689
48	147.000000	3.210487	0.340456
49	148.000000	3.177978	0.337552
50	149.000000	3.085738	0.329309
51	150.000000	2.739086	0.298254

Table V.4.4. (Continued)

	STD. DEV.	CORRELATION 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	4.829802D-01	100																								
2	5.074449D-01	77	100																							
3	5.527170D-01	78	78	100																						
4	6.009126D-01	79	79	80	100																					
5	6.118638D-01	79	79	80	81	100																				
6	6.734839D-01	79	80	81	82	82	100																			
7	7.446974D-01	80	81	82	82	82	83	100																		
8	8.546768D-01	81	82	82	83	83	84	85	100																	
9	8.508957D-01	81	82	82	83	83	84	85	86	100																
10	8.486346D-01	81	82	82	83	83	84	85	86	86	100															
11	9.675167D-01	82	82	83	84	84	85	86	87	87	87	100														
12	9.919949D-01	82	82	83	84	84	85	86	87	87	87	88	100													
13	1.131233D+00	83	83	84	85	85	86	87	88	87	87	88	88	100												
14	1.267415D+00	83	84	84	85	85	86	87	88	88	88	89	89	90	100											
15	1.297272D+00	83	84	85	85	86	86	87	88	88	88	89	89	90	90	100										
16	1.392758D+00	83	84	85	86	86	87	87	88	88	88	89	89	90	91	91	100									
17	1.490564D+00	84	84	85	86	86	87	88	89	89	89	90	90	90	91	91	91	100								
18	1.626450D+00	84	85	85	86	86	87	88	89	89	89	90	90	91	91	91	92	92	100							
19	1.734371D+00	84	85	86	86	87	87	88	89	89	89	90	90	91	92	92	92	92	93	100						
20	1.759746D+00	84	85	86	87	87	88	88	89	89	89	90	90	91	92	92	92	92	93	93	100					
21	1.893845D+00	84	85	86	87	87	88	89	90	90	90	90	91	91	92	92	92	93	93	93	93	100				
22	1.917231D+00	85	85	86	87	87	88	89	90	90	90	90	91	91	92	92	92	93	93	93	93	93	100			
23	1.954147D+00	85	85	86	87	87	88	89	90	90	90	90	91	91	92	92	92	93	93	93	93	93	94	100		
24	1.911941D+00	85	85	86	87	87	88	89	90	90	90	90	91	91	92	92	92	93	93	93	93	93	93	93	100	
25	1.940890D+00	85	85	86	87	87	88	89	90	90	90	90	91	91	92	92	92	93	93	93	93	93	93	94	93	100
26	1.901959D+00	85	85	86	87	87	88	89	90	90	90	90	91	91	92	92	92	93	93	93	93	93	93	93	93	93
27	1.741532D+00	84	85	86	86	87	87	88	89	89	89	90	90	91	92	92	92	92	93	93	93	93	93	93	93	93
28	1.710321D+00	84	85	86	86	87	87	88	89	89	89	90	90	91	91	92	92	92	93	93	93	93	93	93	93	93
29	1.628463D+00	84	85	85	86	86	87	88	89	89	89	90	90	91	91	91	92	92	92	93	93	93	93	93	93	93
30	1.482523D+00	84	84	85	86	86	87	88	89	89	89	90	90	90	91	91	91	92	92	92	92	92	93	93	93	93
31	1.381119D+00	83	84	85	86	86	87	87	88	88	88	89	89	90	91	91	91	91	92	92	92	92	92	92	92	92
32	1.283610D+00	83	84	85	85	86	86	87	88	88	88	89	89	90	90	90	91	91	91	92	92	92	92	92	92	92
33	1.214618D+00	83	83	84	85	85	86	87	88	88	88	89	89	90	90	90	90	91	91	91	91	92	92	92	92	92
34	1.077277D+00	82	83	84	85	85	86	86	87	87	87	88	88	89	89	90	90	90	90	91	91	91	91	91	91	91
35	1.014151D+00	82	83	83	84	84	85	86	87	87	87	88	88	89	89	89	90	90	90	90	90	91	91	91	91	91
36	9.115614D-01	81	82	83	84	84	85	85	86	86	86	87	87	88	88	89	89	89	89	90	90	90	90	90	90	90
37	8.465449D-01	81	82	82	83	83	84	85	86	86	86	87	87	87	88	88	88	89	89	89	89	89	90	90	90	90
38	7.047386D-01	80	80	81	82	82	83	84	85	85	84	85	85	86	87	87	87	87	88	88	88	88	88	88	88	88
39	6.752459D-01	79	80	81	82	82	82	83	84	84	84	85	85	86	86	86	87	87	87	87	88	88	88	88	88	88
40	6.444227D-01	79	80	80	81	81	82	83	84	84	84	85	85	85	86	86	86	87	87	87	87	87	87	87	87	87
41	5.967982D-01	78	79	80	81	81	81	82	83	83	83	84	84	85	85	85	86	86	86	86	86	87	87	87	87	87
42	4.938277D-01	77	77	78	79	79	80	80	81	81	81	82	82	83	83	83	84	84	84	84	85	85	85	85	85	85
43	4.990034D-01	77	77	78	79	79	80	80	81	81	81	82	82	83	83	84	84	84	84	85	85	85	85	85	85	85
44	4.653642D-01	76	77	77	78	78	79	80	81	81	81	81	81	82	83	83	83	83	84	84	84	84	84	84	84	84
45	4.295801D-01	75	76	76	77	77	78	79	80	80	80	80	81	81	82	82	82	82	83	83	83	83	83	83	83	83
46	4.078426D-01	75	75	76	77	77	77	78	79	79	79	80	80	81	81	81	81	82	82	82	82	82	82	83	82	83
47	3.776888D-01	74	74	75	76	76	77	77	78	78	78	79	79	80	80	80	80	81	81	81	81	81	81	82	81	81
48	3.404562D-01	72	73	74	74	74	75	76	77	77	77	77	77	78	79	79	79	79	79	80	80	80	80	80	80	80
49	3.375524D-01	72	73	73	74	74	75	76	77	77	77	77	77	78	78	79	79	79	79	80	80	80	80	80	80	80
50	3.293089D-01	72	72	73	74	74	75	75	76	76	76	77	77	78	78	78	78	79	79	79	79	79	79	80	79	79
51	2.982545D-01	70	71	72	72	72	73	74	75	75	75	75	75	76	77	77	77	77	77	78	78	78	78	78	78	78

Table V.4.4. (Continued)

		26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
26	1.901959D+00	100																								
27	1.741532D+00	93	100																							
28	1.710321D+00	93	93	100																						
29	1.628463D+00	93	93	93	100																					
30	1.482523D+00	92	92	92	92	100																				
31	1.381119D+00	92	92	92	92	91	100																			
32	1.283610D+00	92	92	92	91	91	91	100																		
33	1.214618D+00	92	91	91	91	91	90	90	100																	
34	1.077277D+00	91	91	91	90	90	90	90	89	100																
35	1.014151D+00	91	90	90	90	90	89	89	89	88	100															
36	9.115614D-01	90	90	90	89	89	89	89	88	88	87	100														
37	8.465449D-01	90	89	89	89	89	88	88	88	87	87	86	100													
38	7.047386D-01	88	88	88	88	87	87	87	86	86	86	85	84	100												
39	6.752459D-01	88	88	87	87	87	87	86	86	86	85	85	84	83	100											
40	6.444227D-01	87	87	87	87	87	86	86	86	85	85	84	84	82	82	100										
41	5.967982D-01	87	86	86	86	86	86	85	85	84	84	84	83	82	81	81	100									
42	4.938277D-01	85	84	84	84	84	84	83	83	83	82	82	81	80	80	79	79	100								
43	4.990034D-01	85	85	85	84	84	84	83	83	83	82	82	81	80	80	79	79	77	100							
44	4.653642D-01	84	84	84	84	83	83	83	82	82	82	81	81	79	79	79	78	76	76	100						
45	4.295801D-01	83	83	83	83	82	82	82	82	81	81	80	80	78	78	78	77	75	76	75	100					
46	4.078426D-01	82	82	82	82	82	81	81	81	80	80	79	79	78	78	77	77	75	75	74	73	100				
47	3.776888D-01	81	81	81	81	81	80	80	80	79	79	78	78	77	77	76	76	74	74	73	72	72	100			
48	3.404562D-01	80	80	80	79	79	79	79	78	78	78	77	77	75	75	75	74	73	73	72	71	71	70	100		
49	3.375524D-01	80	80	80	79	79	79	79	78	78	77	77	76	75	75	75	74	72	73	72	71	70	70	68	100	
50	3.293089D-01	79	79	79	79	79	78	78	78	77	77	77	76	75	75	74	74	72	72	72	71	70	69	68	68	100
51	2.982545D-01	78	78	78	77	77	77	77	76	76	76	75	75	73	73	73	72	71	71	70	69	69	68	67	67	66

		51
51	2.982545D-01	100

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	120.000000	123.522867
2	30.000000	23.859774

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	120.000000	123.221543
2	30.000000	22.601910

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	120.000000	123.219438	0.065499
2	30.000000	22.595304	0.189104

	STD. DEV.	CORRELATION 1	2
1	6.549912D-02	100	
2	1.891043D-01	3	100

Table V.4.5. User-Supplied Subroutines for the Second Method of Solution for Example 4. These routines are stored on PDP-10 file EX42.F4.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARAM, PARCOV, AND IFPAR (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C        NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 3
      PARAM(1) = 120.
      PARAM(2) = 30.
      PARAM(3) = 1.14D0
      DO 20 I=1,NPARAM
           DO 10 L=1,I
                PARCOV(L,I) = 0.D0
                IF (L.EQ.I) PARCOV(L,I) = PARAM(I)**2
   10      CONTINUE
   20 CONTINUE
      PARCOV(NPARAM,NPARAM) = 0.01D0
      RETURN
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
C        NML, MARCH 1981
C
```

Table V.4.5. (Continued)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME)
C
C *** READ FILE 24 TO COUNT THE NUMBER OF DATA POINTS (NDAT)
      N = 0
   10 READ (24,99997,END=20) EE, D, ERR
      IF (EE.LE.0.DO.AND.D.LE.DO) GO TO 20
      N = N + 1
      GO TO 10
   20 CONTINUE
      NDAT = N
      REWIND 24
C
C
      DO 30 I=1,NDAT
           READ (24,99997) EE, D, ERR
           E(I) = EE
           DATA(I) = D
           E2(I) = ERR
   30 CONTINUE
      CLOSE (UNIT=24)
C
      DO 50 I=1,NDAT
           DO 40 L=1,I
                DATCOV(L,I) = 0.DO
                IF (L.EQ.I) DATCOV(L,I) = E2(I)**2
   40      CONTINUE
   50 CONTINUE
      RETURN
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (3F12.1)
      END
```

Table V.4.5. (Continued)

```
C
C
C
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C        NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DATA IN /0/
      IF (IN.GT.0) GO TO 10
      AM = 100.
      CAY = 2.1968E-4*(AM/(AM+1.009))
      CAYS = 0.0314159/CAY**2/1000.
      IN = 1
   10 EEE = E(KDAT)
      THEO = (CAYS/EEE)*PARAM(2)**2/((EEE-PARAM(1))**2+0.25*
     *     PARAM(2)**2)
      THEO = THEO/PARAM(3)
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR PARAMETER KPAR,
C ***            AT ENERGY E(KDAT)
C
C        NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DATA IN /0/
      IF (IN.GT.0) GO TO 10
      AM = 100.
      CAY = 2.1968E-4*(AM/(AM+1.009))
      CAYS = 0.0314159/CAY**2/1000.
      IN = 1
C
```

Table V.4.5. (Continued)

```
10 EEE = E(KDAT)
   DELE = EEE - PARAM(1)
   D = DELE**2 + 0.25*PARAM(2)**2
   B = (CAYS/EEE)*PARAM(2)/(D*PARAM(3))
   D = 2.0D0*B*DELE/D
   I = KPAR
   GO TO (20, 30, 40), I
20 GG = D*PARAM(2)
   GO TO 50
30 GG = D*DELE
   GO TO 50
40 GG = -B*PARAM(2)/PARAM(3)
50 CONTINUE
   DERIV = GG
   RETURN
   END
```

Table V.4.6. Teletype Commands and User Responses to Program Prompts for Example 4, Second Method of Solution.

```
.EX BAYES,EX42,SHOW,DPPXX,DO4NML ↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001 ↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N ↓
WHAT'S NAME OF THE DATA FILE? EX41.DAT ↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.4.7. Output Resulting from the Use of Commands Given in Table V.4.6.

```
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS    0.001000

******INPUT PARAMETER VALUES

        PARAMETER       UNCERTAINTY
   1     120.000000     120.000000
   2      30.000000      30.000000
   3       1.140000       0.100000

***** INTERMEDIATE RESULTS

        OLD PARAM        NEW PARAMETERS
   1     120.000000     123.320330
   2      30.000000      24.326760
   3       1.140000       1.207887

***** INTERMEDIATE RESULTS

        OLD PARAM        NEW PARAMETERS
   1     120.000000     123.216831
   2      30.000000      22.522077
   3       1.140000       1.134500

***** INTERMEDIATE RESULTS

        OLD PARAM        NEW PARAMETERS
   1     120.000000     123.218900
   2      30.000000      22.643774
   3       1.140000       1.137689

******* NEW VALUES FOR PARAMETERS

           POLD           PNEW        UNCERTAINTY
   1     120.000000     123.219184       0.065389
   2      30.000000      22.642567       0.188848
   3       1.140000       1.137635       0.008195

       STD. DEV.     CORRELATION

                     1   2   3
  1  6.538899D-02 100
  2  1.888479D-01   3 100
  3  8.195363D-03  -2  78 100
```

Table V.4.8. Teletype Commands and User Responses to Program Prompts for Example 4, Second Method of Solution, Where Subsets of the Data are Analyzed Sequentially.

```
.EX BAYES,EX42,SHOW,DPPXX,DO4NML ↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 5,.001 ↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N ↓
WHAT'S NAME OF THE DATA FILE? EX41A.DAT ↓
WHAT'S NAME OF THE DATA FILE? EX41B.DAT ↓
WHAT'S NAME OF THE DATA FILE? ↓
STOP
```

Table V.4.9. Output Resulting from the Use of Commands Given in Table V.4.8.

MAXIMUM NUMBER OF ITERATIONS IS 5
CONVERGENCE FACTOR IS 0.001000

******INPUT PARAMETER VALUES

	PARAMETER	UNCERTAINTY
1	120.000000	120.000000
2	30.000000	30.000000
3	1.140000	0.100000

****** INPUT DATA VALUES

	DATA POINT	VALUE	UNCERTAINTY
1	100.000000	4.223518	0.205512
2	101.000000	4.466085	0.211331
3	102.000000	4.915435	0.221708
4	103.000000	5.394355	0.232258
5	104.000000	5.503246	0.234590
6	105.000000	6.116313	0.247312
7	106.000000	6.825470	0.261256
8	107.000000	7.921634	0.281454
9	108.000000	7.883934	0.280783
10	109.000000	7.861386	0.280382
11	110.000000	9.047163	0.300785
12	111.000000	9.291409	0.304818
13	112.000000	10.681179	0.326821
14	113.000000	12.041003	0.347001
15	114.000000	12.339183	0.351272
16	115.000000	13.292924	0.364595
17	116.000000	14.269995	0.377756
18	117.000000	15.627672	0.395319
19	118.000000	16.706080	0.408731
20	119.000000	16.959656	0.411821
21	120.000000	18.299793	0.427783
22	121.000000	18.533516	0.430506
23	122.000000	18.902473	0.434770
24	123.000000	18.480649	0.429891
25	124.000000	18.769973	0.433243
26	125.000000	18.380886	0.428729

Table V.4.9. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	120.000000	124.211134
2	30.000000	26.918133
3	1.140000	1.218049

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	120.000000	122.560816
2	30.000000	21.342463
3	1.140000	1.147678

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	120.000000	122.923917
2	30.000000	22.193190
3	1.140000	1.142256

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	120.000000	122.944974
2	30.000000	22.202865
3	1.140000	1.140846

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	120.000000	122.945288	0.220635
2	30.000000	22.203036	0.355430
3	1.140000	1.140832	0.010947

	STD. DEV.	CORRELATION		
		1	2	3
1	2.206350D-01	100		
2	3.554301D-01	75	100	
3	1.094666D-02	-48	6	100

Table V.4.9. (Continued)

****** INPUT DATA VALUES

	DATA POINT	VALUE	UNCERTAINTY
1	126.000000	16.777637	0.409605
2	127.000000	16.465754	0.405780
3	128.000000	15.647785	0.395573
4	129.000000	14.189659	0.376692
5	130.000000	13.176671	0.362997
6	131.000000	12.202733	0.349324
7	132.000000	11.513750	0.339319
8	133.000000	10.142552	0.318474
9	134.000000	9.512503	0.308423
10	135.000000	8.488945	0.291358
11	136.000000	7.840551	0.280010
12	137.000000	6.427479	0.253525
13	138.000000	6.133853	0.247666
14	139.000000	5.827105	0.241394
15	140.000000	5.353455	0.231375
16	141.000000	4.331043	0.208112
17	142.000000	4.382364	0.209341
18	143.000000	4.048990	0.201221
19	144.000000	3.694872	0.192220
20	145.000000	3.480069	0.186549
21	146.000000	3.182577	0.178398
22	147.000000	2.816217	0.167816
23	148.000000	2.787700	0.166964
24	149.000000	2.706788	0.164523
25	150.000000	2.402707	0.155007

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	122.945288	123.215984
2	22.203036	22.656192
3	1.140832	1.138229

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	122.945288	123.217381	0.064962
2	22.203036	22.644752	0.186979
3	1.140832	1.137768	0.008168

	STD. DEV.	CORRELATION 1	2	3
1	6.496191D-02	100		
2	1.869789D-01	5	100	
3	8.168167D-03	1	78	100

Example 5. Angular Distribution

The Problem:

Find the coefficients in a Legendre expansion of an angular distribution which was generated from a physical model.

The Solution:

The theoretical values are given by the Legendre expansion

$$T(\theta) = \sum_{\ell=0}^{k} \sqrt{\frac{2\ell+1}{4\pi}} \; P_\ell \; Y_{\ell 0}(\cos\theta) \qquad \text{(V.5.1)}$$

where $Y_{\ell 0}(x)$ is the spherical harmonic with m=0, equal to $\sqrt{\frac{2\ell+1}{4\pi}}$ times the Legendre polynomial of degree ℓ. This expression is linear with respect to parameters P_ℓ, the partial derivatives being given by $\sqrt{\frac{2\ell+1}{4\pi}}\, Y_{\ell 0}(\cos\theta)$.

Because the "data" points D_i represent a calculated differential cross section, they are not independent. Rather, the integrated cross section is constant:

$$\sigma = \int \frac{d\sigma}{d\Omega}\, d\Omega = \int_0^{2\pi} d\phi \int_{-1}^{1} \frac{d\sigma}{d\Omega}(\theta)\, d(\cos\theta) \qquad \text{(V.5.2)}$$

Replacing the integral by a summation and $\frac{d\sigma}{d\Omega}(\theta_i)$ by D_i gives

$$\sigma = 2\pi \sum_i D_i \sin(\theta_i)\, (\Delta\theta)_i \qquad \text{(V.5.3)}$$

This is a mathematical expression of the physical fact that the particle must appear at one, and only one, value of θ.

Small changes in data D_i result in no change in σ:

$$0 = \delta\sigma = 2\pi \sum_i \delta D_i \sin\theta_i \Delta\theta_i \tag{V.5.4}$$

Multiplying by δD_j and taking the expectation value give the following relationship for the covariance matrix elements $V_{ij} = \langle \delta D_i \ \delta D_j \rangle$:

$$0 = \sum_i V_{ij} \sin\theta_i \Delta\theta_i \tag{V.5.5}$$

It is easy to see that the expression

$$V_{ij} = (\Delta D_i)^2 \delta_{ij} - \frac{(\Delta D_i)^2 \sin\theta_i \Delta\theta_i \Delta\theta_j \sin\theta_j (\Delta D_j)^2}{\sum_k (\Delta D_k \sin\theta_k \Delta\theta_k)^2} \tag{V.5.6}$$

satisfies the relationship (V.5.5), where ΔD_i is the measured uncertainty on D_i. In Appendix A we demonstrate that this is in fact the only correct form. Note that this matrix is singular and thus has no inverse; least-squares weighted residuals and χ^2 must therefore not be evaluated here.

Specific Case

The "experimental data" used here are theoretical points resulting from a GENØA [PE67] fit to natural Calcium angular distribution data [HE82]. An arbitrary 5% uncertainty ΔD_i was assigned to each of the 44 data points D_i.

The data are listed in Table V.5.1. Starting values for the 21 parameters are given in Table V.5.2. User-supplied subroutines are shown in Table V.5.3. Execution commands are given in Table V.5.4. and output shown in Table V.5.5. No iteration is required for this problem, since the function shown in Eq. (V.5.1) is linear in the parameters P_ℓ; nevertheless we have specified one iteration in order to obtain a listing of the theoretical cross section evaluated at the new parameter values.

Table V.5.6. shows the results which would be obtained if the data covariance matrix were naively set equal to

$$V_{ij} = (\Delta D_i)^2 \; \delta_{ij}.$$

Note that the differences between data and theory are more pronounced here, especially at forward angles.

Table V.5.1. ØRELA PDP-10 File EX5.DAT, Containing Data for Example 5.

Angle in Degrees		Data	Relative Uncertainty
0.4000E+02			
0.0000E+00	0.0000E+00	0.6118E+04	0.5000E+01
0.4000E+01	0.0000E+00	0.5665E+04	0.5000E+01
0.8000E+01	0.0000E+00	0.4481E+04	0.5000E+01
0.1200E+02	0.0000E+00	0.2992E+04	0.5000E+01
0.1600E+02	0.0000E+00	0.1651E+04	0.5000E+01
0.2000E+02	0.0000E+00	0.7322E+03	0.5000E+01
0.2400E+02	0.0000E+00	0.2695E+03	0.5000E+01
0.2800E+02	0.0000E+00	0.1280E+03	0.5000E+01
0.3200E+02	0.0000E+00	0.1303E+03	0.5000E+01
0.3600E+02	0.0000E+00	0.1504E+03	0.5000E+01
0.4000E+02	0.0000E+00	0.1393E+03	0.5000E+01
0.4400E+02	0.0000E+00	0.1019E+03	0.5000E+01
0.4800E+02	0.0000E+00	0.6055E+02	0.5000E+01
0.5200E+02	0.0000E+00	0.3128E+02	0.5000E+01
0.5600E+02	0.0000E+00	0.1723E+02	0.5000E+01
0.6000E+02	0.0000E+00	0.1328E+02	0.5000E+01
0.6400E+02	0.0000E+00	0.1280E+02	0.5000E+01
0.6800E+02	0.0000E+00	0.1174E+02	0.5000E+01
0.7200E+02	0.0000E+00	0.9224E+01	0.5000E+01
0.7600E+02	0.0000E+00	0.6131E+01	0.5000E+01
0.8000E+02	0.0000E+00	0.3578E+01	0.5000E+01
0.8400E+02	0.0000E+00	0.2070E+01	0.5000E+01
0.8800E+02	0.0000E+00	0.1470E+01	0.5000E+01
0.9200E+02	0.0000E+00	0.1346E+01	0.5000E+01
0.9600E+02	0.0000E+00	0.1305E+01	0.5000E+01
0.1000E+03	0.0000E+00	0.1160E+01	0.5000E+01
0.1040E+03	0.0000E+00	0.9004E+00	0.5000E+01
0.1080E+03	0.0000E+00	0.6101E+00	0.5000E+01
0.1120E+03	0.0000E+00	0.3723E+00	0.5000E+01
0.1160E+03	0.0000E+00	0.2269E+00	0.5000E+01
0.1200E+03	0.0000E+00	0.1672E+00	0.5000E+01
0.1240E+03	0.0000E+00	0.1599E+00	0.5000E+01
0.1280E+03	0.0000E+00	0.1687E+00	0.5000E+01
0.1320E+03	0.0000E+00	0.1695E+00	0.5000E+01
0.1360E+03	0.0000E+00	0.1534E+00	0.5000E+01
0.1400E+03	0.0000E+00	0.1235E+00	0.5000E+01
0.1440E+03	0.0000E+00	0.8725E-01	0.5000E+01
0.1480E+03	0.0000E+00	0.5286E-01	0.5000E+01
0.1520E+03	0.0000E+00	0.2753E-01	0.5000E+01
0.1560E+03	0.0000E+00	0.1780E-01	0.5000E+01
0.1600E+03	0.0000E+00	0.2879E-01	0.5000E+01
0.1640E+03	0.0000E+00	0.6171E-01	0.5000E+01
0.1680E+03	0.0000E+00	0.1112E+00	0.5000E+01
0.1720E+03	0.0000E+00	0.1648E+00	0.5000E+01
0.1760E+03	0.0000E+00	0.2063E+00	0.5000E+01
0.1800E+03	0.0000E+00	0.2220E+00	0.5000E+01

Table V.5.2. Starting Values for the Parameters for Example 5, Stored in File EX5.PAR.

```
1300. 1000. 800. 600. 400. 200. 150. 100.
90. 80. 70. 60. 55. 50. 45. 40.
35. 30. 25. 20. 18. 15. 10. 8.
5.
```

Table V.5.3. User-Supplied Subroutines for Example 5, Stored in File EX5.F4.

```
C
C
C
      SUBROUTINE SETPAR
C
C *** PURPOSE -- INITIALIZE ARRAYS PARM, VRPR, AND IFLAG (INITIAL
C ***            VALUES FOR PARAMETERS, COVARIANCE MATRIX FOR
C ***            PARAMETERS, AND FLAG IF VARIED, RESPECTIVELY)
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      NPARAM = 21
        OPEN (UNIT=19,FILE='EX5.PAR')
      READ (19,99998) (PARAM(I),I=1,NPARAM)
C
      DO 40 I=1,NPARAM
           DO 30 L=1,I
                PARCOV(L,I) = 0.D0
                IF (L.EQ.I) PARCOV(L,I) = 0.01*PARAM(I)**2
   30      CONTINUE
   40 CONTINUE
      RETURN
C
99999 FORMAT (33H INITIAL GUESSES FOR PARAMETERS? $)
99998 FORMAT (8F/8F/8F/8F)
99997 FORMAT (27H WHICH PARAMETERS TO VARY? $)
99996 FORMAT (20I)
      END
C
C
C
      SUBROUTINE SETDAT
C
C *** PURPOSE -- INPUT NDAT (THE NUMBER OF DATA POINTS),
C ***            INPUT THE ARRAYS E (ENERGY), DATA (EXPERIMENTAL
C ***            DATA), AND DATCOV (COVARIANCE MATRIX OF THE
C ***            EXPERIMENTAL DATA).
C
```

Table V.5.3. (Continued)

```
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NAME, DBLANK
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
      DATA DBLANK /10H          /, PI180/0.017453293D0/
C
      WRITE (5,99999)
      READ (5,99998) NAME
      IF (NAME.EQ.DBLANK) STOP
      OPEN (UNIT=24, FILE=NAME, ACCESS='SEQIN')
C
C
        NDAT=44
C
C *** INITIALIZE DATA COVARIANCE MATRIX DATCOV
      DO 40 I=1,NDAT
           DO 30 L=1,I
                DATCOV(L,I) = 0.D0
   30      CONTINUE
   40 CONTINUE
C
C *** READ FILE 24 TO OBTAIN ENERGY, DATA, AND UNCERTAINTIES,
C ***   SET COVARIANCE MATRIX
        READ(24,99)
99      FORMAT (1X)
        SUM=0.0
      DO 50 I=1,NDAT
           READ (24,99997) EE,ABCD, D, ERR
           E(I) = EE
           DATA(I) = D
           ERR=.01
           DATCOV(I,I)=(ERR*D)**2
           EEEEE=(ERR*D*DSIN(EE*PI180))**2
           SUM=SUM+EEEEE
           E2(I)=(ERR*D)**2*DSIN(EE*PI180)
   50 CONTINUE
      CLOSE (UNIT=24)
```

Table V.5.3. (Continued)

```
C
      DO 70 I=1,NDAT
           DO 60 J=1,NDAT
                DATCOV(J,I)=DATCOV(J,I)-E2(I)*E2(J)/SUM
60         CONTINUE
70    CONTINUE
      RETURN
C
99999 FORMAT (31H WHAT'S NAME OF THE DATA FILE? $)
99998 FORMAT (A10)
99997 FORMAT (4E11.4)
      END
C
C
C
      DOUBLE PRECISION FUNCTION THEO(KDAT)
C
C *** PURPOSE -- GENERATE THEORETICAL VALUE OF THE FUNCTION (EG
C ***            CROSS SECTION) AT DATA POINT KDAT AND PARAMETERS PARAM
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
        COMMON/DCL/PL(26)
        EE=E(KDAT)
        CANC=DCOS(EE*.017453292D0)
        PL(1)=1.0D0
        SIGG=1.0D0
        NP=NPARAM
        LM=NP+1
        PL(2)=CANC
        P2=PL(2)
        C1=3.0D0
        C2=1.0D0
        C3=2.0D0
        DO 4 I=3,LM
        PL(I)=(C1*P2*PL(I-1)-C2*PL(I-2))/C3
        C1=C1+2.0D0
        C2=C2+1.0D0
        C3=C3+1.0D0
```

Table V.5.3. (Continued)

```
4       CONTINUE
        YCU=0.0D0
        C1=0.5D0
        DO 5 I=1,NP
        YCU=(YCU+C1*PARAM(I)*PL(I))
        C1=C1+1.0D0
5       CONTINUE
        THEO=YCU*SIGG/6.28318D0
      RETURN
      END
C
C
C
      DOUBLE PRECISION FUNCTION DERIV(KDAT,KPAR)
C
C *** PURPOSE -- GENERATE DERIVATIVES FOR ALL PARAMETERS,
C ***            AT ENERGY E(KDAT)
C
C       NML, MARCH 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
        COMMON/DCL/PL(26)
C
        SIGG=1.0D0
        DERIV=PL(KPAR)*(2.D0*KPAR-1.D0)*0.5D0*SIGG/6.28318D0
      RETURN
      END
```

Table V.5.4. Teletype Commands and User Responses to Program Prompts for Example 5.

```
.EX BAYES.EX5.SHOW.DPPXX.DO4NML ↓
HOW MANY ITERATIONS?  WHAT IS CONVERGENCE FRACTION? 1,0. ↓
DO YOU WISH TO USE AUTOMATIC NUMERICAL DERIVATIVES? N ↓
WHAT'S NAME OF THE DATA FILE? EX5.DAT ↓
WHAT'S NAME OF THE DATA FILE?  ↓
STOP
```

Table V.5.5. Output Resulting from the Use of Commands Given in Table V.5.4.

MAXIMUM NUMBER OF ITERATIONS IS 1
CONVERGENCE FACTOR IS 0.000000

******INPUT PARAMETER VALUES

	PARAMETER	UNCERTAINTY
1	1300.000000	130.000000
2	1000.000000	100.000000
3	800.000000	80.000000
4	600.000000	60.000000
5	400.000000	40.000000
6	200.000000	20.000000
7	150.000000	15.000000
8	100.000000	10.000000
9	90.000000	9.000000
10	80.000000	8.000000
11	70.000000	7.000000
12	60.000000	6.000000
13	55.000000	5.500000
14	50.000000	5.000000
15	45.000000	4.500000
16	40.000000	4.000000
17	35.000000	3.500000
18	30.000000	3.000000
19	25.000000	2.500000
20	20.000000	2.000000
21	18.000000	1.800000

****** INPUT DATA VALUES

	DATA POINT	VALUE	UNCERTAINTY
1	0.000000	6118.000000	61.180000
2	4.000000	5665.000000	52.982656
3	8.000000	4481.000000	37.167439
4	12.000000	2992.000000	24.845156
5	16.000000	1651.000000	15.076195
6	20.000000	732.199997	7.135425
7	24.000000	269.500000	2.681979
8	28.000000	128.000000	1.278145
9	32.000000	130.299999	1.300506
10	36.000000	150.400000	1.499278
11	40.000000	139.299999	1.388513
12	44.000000	101.900000	1.016950
13	48.000000	60.550000	0.605008
14	52.000000	31.280000	0.312724

Table V.5.5. (Continued)

15	56.000000	17.230000	0.172286
16	60.000000	13.280000	0.132793
17	64.000000	12.800000	0.127993
18	68.000000	11.740000	0.117394
19	72.000000	9.224000	0.092237
20	76.000000	6.131000	0.061309
21	80.000000	3.578000	0.035780
22	84.000000	2.070000	0.020700
23	88.000000	1.470000	0.014700
24	92.000000	1.346000	0.013460
25	96.000000	1.305000	0.013050
26	100.000000	1.160000	0.011600
27	104.000000	0.900400	0.009004
28	108.000000	0.610100	0.006101
29	112.000000	0.372300	0.003723
30	116.000000	0.226900	0.002269
31	120.000000	0.167200	0.001672
32	124.000000	0.159900	0.001599
33	128.000000	0.168700	0.001687
34	132.000000	0.169500	0.001695
35	136.000000	0.153400	0.001534
36	140.000000	0.123500	0.001235
37	144.000000	0.087250	0.000872
38	148.000000	0.052860	0.000529
39	152.000000	0.027530	0.000275
40	156.000000	0.017800	0.000178
41	160.000000	0.028790	0.000288
42	164.000000	0.061710	0.000617
43	168.000000	0.111200	0.001112
44	172.000000	0.164800	0.001648

Table V.5.5. (Continued)

STD. DEV. CORRELATION

	STD. DEV.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	6.118000D+01	100																								
2	5.298266D+01	0	100																							
3	3.716744D+01	0	-25	100																						
4	2.484516D+01	0	-25	-45	100																					
5	1.507620D+01	0	-17	-30	-30	100																				
6	7.135425D+00	0	-9	-16	-15	-10	100																			
7	2.681979D+00	0	-4	-7	-7	-4	-2	100																		
8	1.278145D+00	0	-2	-4	-4	-2	-1	-1	100																	
9	1.300506D+00	0	-2	-4	-4	-3	-1	-1	0	100																
10	1.499278D+00	0	-3	-5	-5	-4	-2	-1	0	0	100															
11	1.388513D+00	0	-3	-5	-5	-4	-2	-1	0	0	-1	100														
12	1.016950D+00	0	-2	-4	-4	-3	-1	-1	0	0	-1	-1	100													
13	6.050080D-01	0	-2	-3	-3	-2	-1	0	0	0	0	0	0	100												
14	3.127238D-01	0	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	100											
15	1.722859D-01	0	0	-1	-1	-1	0	0	0	0	0	0	0	0	0	100										
16	1.327930D-01	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	100									
17	1.279932D-01	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	100								
18	1.173944D-01	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	100							
19	9.223715D-02	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100						
20	6.130913D-02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100					
21	3.577982D-02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100				
22	2.069996D-02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100			
23	1.469999D-02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100		
24	1.345999D-02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
25	1.304999D-02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
26	1.159999D-02	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
27	9.003997D-03	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
28	6.100999D-03	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
29	3.723000D-03	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
30	2.269000D-03	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
31	1.672000D-03	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
32	1.599000D-03	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
33	1.687000D-03	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
34	1.695000D-03	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
35	1.534000D-03	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
36	1.235000D-03	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
37	8.725000D-04	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
38	5.286000D-04	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
39	2.753000D-04	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
40	1.780000D-04	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
41	2.879000D-04	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
42	6.171000D-04	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
43	1.112000D-03	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
44	1.648000D-03	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

		26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
26	1.159999D-02	100																		

Table V.5.5. (Continued)

***** THEORETICAL CALCULATION

	ANGLE	DATA	THEORY
1	0.000000	6118.000000	2989.568980
2	4.000000	5665.000000	2667.003150
3	8.000000	4481.000000	1936.630650
4	12.000000	2992.000000	1269.171320
5	16.000000	1651.000000	928.462798
6	20.000000	732.199997	820.155669
7	24.000000	269.500000	724.424120
8	28.000000	128.000000	563.431033
9	32.000000	130.299999	407.382153
10	36.000000	150.400000	311.634125
11	40.000000	139.299999	245.797290
12	44.000000	101.900000	169.219584
13	48.000000	60.550000	95.186277
14	52.000000	31.280000	53.413273
15	56.000000	17.230000	36.346632
16	60.000000	13.280000	17.327535
17	64.000000	12.800000	-3.989127
18	68.000000	11.740000	-8.245897
19	72.000000	9.224000	4.390171
20	76.000000	6.131000	12.807627
21	80.000000	3.578000	9.517585
22	84.000000	2.070000	9.250757
23	88.000000	1.470000	19.822928
24	92.000000	1.346000	28.370874
25	96.000000	1.305000	25.136642
26	100.000000	1.160000	20.735484
27	104.000000	0.900400	26.771720
28	108.000000	0.610100	34.961570
29	112.000000	0.372300	31.459558
30	116.000000	0.226900	20.716708
31	120.000000	0.167200	17.087411
32	124.000000	0.159900	19.486326
33	128.000000	0.168700	13.984942
34	132.000000	0.169500	-0.169745
35	136.000000	0.153400	-7.322895
36	140.000000	0.123500	-1.671079
37	144.000000	0.087250	2.752042
38	148.000000	0.052860	-2.910797
39	152.000000	0.027530	-5.246524
40	156.000000	0.017800	9.094583
41	160.000000	0.028790	28.454175
42	164.000000	0.061710	33.422376
43	168.000000	0.111200	30.304037
44	172.000000	0.164800	43.239372

Table V.5.5. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	1300.000000	1321.849700
2	1000.000000	1226.239260
3	800.000000	1078.113730
4	600.000000	921.247207
5	400.000000	779.398097
6	200.000000	658.724348
7	150.000000	556.748930
8	100.000000	467.775757
9	90.000000	386.392693
10	80.000000	308.638560
11	70.000000	233.219483
12	60.000000	161.841183
13	55.000000	100.224253
14	50.000000	54.079274
15	45.000000	24.740608
16	40.000000	9.029796
17	35.000000	2.144524
18	30.000000	-0.079910
19	25.000000	-0.379305
20	20.000000	-0.185724
21	18.000000	-0.038155

***** THEORETICAL CALCULATION

	ANGLE	DATA	THEORY
1	0.000000	6118.000000	5683.096230
2	4.000000	5665.000000	5298.620450
3	8.000000	4481.000000	4277.620880
4	12.000000	2992.000000	2953.220210
5	16.000000	1651.000000	1704.105300
6	20.000000	732.199997	796.223274
7	24.000000	269.500000	303.478916
8	28.000000	128.000000	134.635478
9	32.000000	130.299999	129.051935
10	36.000000	150.400000	152.883642
11	40.000000	139.299999	144.937853
12	44.000000	101.900000	106.138509
13	48.000000	60.550000	61.881116
14	52.000000	31.280000	31.156701
15	56.000000	17.230000	17.167042
16	60.000000	13.280000	13.486200
17	64.000000	12.800000	12.919632
18	68.000000	11.740000	11.692422
19	72.000000	9.224000	9.169609
20	76.000000	6.131000	6.154340

Table V.5.5. (Continued)

21	80.000000	3.578000	3.614149
22	84.000000	2.070000	2.070820
23	88.000000	1.470000	1.459918
24	92.000000	1.346000	1.350166
25	96.000000	1.305000	1.314820
26	100.000000	1.160000	1.158798
27	104.000000	0.900400	0.894858
28	108.000000	0.610100	0.609327
29	112.000000	0.372300	0.374506
30	116.000000	0.226900	0.227078
31	120.000000	0.167200	0.166276
32	124.000000	0.159900	0.160360
33	128.000000	0.168700	0.169465
34	132.000000	0.169500	0.168931
35	136.000000	0.153400	0.152752
36	140.000000	0.123500	0.123735
37	144.000000	0.087250	0.087666
38	148.000000	0.052860	0.052764
39	152.000000	0.027530	0.027439
40	156.000000	0.017800	0.017875
41	160.000000	0.028790	0.028649
42	164.000000	0.061710	0.061781
43	168.000000	0.111200	0.111780
44	172.000000	0.164800	0.164213

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	1300.000000	1321.849700	0.047574
2	1000.000000	1226.239260	0.204550
3	800.000000	1078.113730	0.475384
4	600.000000	921.247207	0.745865
5	400.000000	779.398097	0.944152
6	200.000000	658.724348	1.046241
7	150.000000	556.748930	1.073299
8	100.000000	467.775757	1.069676
9	90.000000	386.392693	1.066781
10	80.000000	308.638560	1.065662
11	70.000000	233.219483	1.051189
12	60.000000	161.841183	1.009715
13	55.000000	100.224253	0.935891
14	50.000000	54.079274	0.830810
15	45.000000	24.740608	0.699623
16	40.000000	9.029796	0.550392
17	35.000000	2.144524	0.394854
18	30.000000	-0.079910	0.248038
19	25.000000	-0.379305	0.127735
20	20.000000	-0.185724	0.047709
21	18.000000	-0.038155	0.009775

Table V.5.5. (Continued)

	STD. DEV.	CORRELATION																				
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	4.757400D-02	100																				
2	2.045496D-01	70	100																			
3	4.753837D-01	65	99	100																		
4	7.458649D-01	68	99	99	100																	
5	9.441518D-01	73	97	97	99	100																
6	1.046241D+00	80	93	93	96	99	100															
7	1.073299D+00	88	87	86	89	94	98	100														
8	1.069676D+00	93	78	75	79	85	92	98	100													
9	1.066781D+00	95	68	64	68	74	83	92	98	100												
10	1.065662D+00	95	60	55	58	65	74	84	93	99	100											
11	1.051189D+00	95	56	50	53	59	68	79	89	96	99	100										
12	1.009715D+00	96	55	49	52	58	66	76	85	92	97	99	100									
13	9.358908D-01	97	57	51	53	58	66	75	83	89	94	97	99	100								
14	8.308099D-01	97	58	52	54	59	66	74	81	86	90	93	96	99	100							
15	6.996233D-01	95	57	51	54	59	65	72	78	82	85	88	92	96	99	100						
16	5.503923D-01	92	54	48	51	56	62	69	74	77	79	81	85	90	95	99	100					
17	3.948541D-01	87	49	43	46	51	57	63	68	70	72	74	78	83	89	94	98	100				
18	2.480382D-01	80	42	37	40	45	51	57	61	63	64	66	69	74	80	87	94	98	100			
19	1.277350D-01	71	35	31	33	38	44	49	54	55	56	57	60	65	71	79	86	93	98	100		
20	4.770879D-02	63	29	25	27	31	36	42	46	48	48	49	51	55	61	69	77	86	93	98	100	
21	9.775135D-03	53	23	19	21	25	30	35	38	40	40	41	43	46	51	58	67	76	84	92	98	100

Table V.5.6. Results Which are Obtained for Example 5 by Assuming That the Data Covariance Matrix is Diagonal.

MAXIMUM NUMBER OF ITERATIONS IS 1
CONVERGENCE FACTOR IS 0.000000

******INPUT PARAMETER VALUES

	PARAMETER	UNCERTAINTY
1	1300.000000	130.000000
2	1000.000000	100.000000
3	800.000000	80.000000
4	600.000000	60.000000
5	400.000000	40.000000
6	200.000000	20.000000
7	150.000000	15.000000
8	100.000000	10.000000
9	90.000000	9.000000
10	80.000000	8.000000
11	70.000000	7.000000
12	60.000000	6.000000
13	55.000000	5.500000
14	50.000000	5.000000
15	45.000000	4.500000
16	40.000000	4.000000
17	35.000000	3.500000
18	30.000000	3.000000
19	25.000000	2.500000
20	20.000000	2.000000
21	18.000000	1.800000

****** INPUT DATA VALUES

	DATA POINT	VALUE	UNCERTAINTY
1	0.000000	6118.000000	61.180000
2	4.000000	5665.000000	56.650000
3	8.000000	4481.000000	44.810000
4	12.000000	2992.000000	29.920000
5	16.000000	1651.000000	16.510000
6	20.000000	732.199997	7.322000
7	24.000000	269.500000	2.695000
8	28.000000	128.000000	1.280000
9	32.000000	130.299999	1.303000
10	36.000000	150.400000	1.504000
11	40.000000	139.299999	1.393000
12	44.000000	101.900000	1.019000
13	48.000000	60.550000	0.605500
14	52.000000	31.280000	0.312800

Table V.5.6. (Continued)

15	56.000000	17.230000	0.172300
16	60.000000	13.280000	0.132800
17	64.000000	12.800000	0.128000
18	68.000000	11.740000	0.117400
19	72.000000	9.224000	0.092240
20	76.000000	6.131000	0.061310
21	80.000000	3.578000	0.035780
22	84.000000	2.070000	0.020700
23	88.000000	1.470000	0.014700
24	92.000000	1.346000	0.013460
25	96.000000	1.305000	0.013050
26	100.000000	1.160000	0.011600
27	104.000000	0.900400	0.009004
28	108.000000	0.610100	0.006101
29	112.000000	0.372300	0.003723
30	116.000000	0.226900	0.002269
31	120.000000	0.167200	0.001672
32	124.000000	0.159900	0.001599
33	128.000000	0.168700	0.001687
34	132.000000	0.169500	0.001695
35	136.000000	0.153400	0.001534
36	140.000000	0.123500	0.001235
37	144.000000	0.087250	0.000872
38	148.000000	0.052860	0.000529
39	152.000000	0.027530	0.000275
40	156.000000	0.017800	0.000178
41	160.000000	0.028790	0.000288
42	164.000000	0.061710	0.000617
43	168.000000	0.111200	0.001112
44	172.000000	0.164800	0.001648

Table V.5.6. (Continued)

***** THEORETICAL CALCULATION

	ANGLE	DATA	THEORY
1	0.000000	6118.000000	2989.568980
2	4.000000	5665.000000	2667.003150
3	8.000000	4481.000000	1936.630650
4	12.000000	2992.000000	1269.171320
5	16.000000	1651.000000	928.462798
6	20.000000	732.199997	820.155669
7	24.000000	269.500000	724.424120
8	28.000000	128.000000	563.431033
9	32.000000	130.299999	407.382153
10	36.000000	150.400000	311.634125
11	40.000000	139.299999	245.797290
12	44.000000	101.900000	169.219584
13	48.000000	60.550000	95.186277
14	52.000000	31.280000	53.413273
15	56.000000	17.230000	36.346632
16	60.000000	13.280000	17.327535
17	64.000000	12.800000	-3.989127
18	68.000000	11.740000	-8.245897
19	72.000000	9.224000	4.390171
20	76.000000	6.131000	12.807627
21	80.000000	3.578000	9.517585
22	84.000000	2.070000	9.250757
23	88.000000	1.470000	19.822928
24	92.000000	1.346000	28.370874
25	96.000000	1.305000	25.136642
26	100.000000	1.160000	20.735484
27	104.000000	0.900400	26.771720
28	108.000000	0.610100	34.961570
29	112.000000	0.372300	31.459558
30	116.000000	0.226900	20.716708
31	120.000000	0.167200	17.087411
32	124.000000	0.159900	19.486326
33	128.000000	0.168700	13.984942
34	132.000000	0.169500	-0.169745
35	136.000000	0.153400	-7.322895
36	140.000000	0.123500	-1.671079
37	144.000000	0.087250	2.752042
38	148.000000	0.052860	-2.910797
39	152.000000	0.027530	-5.246524
40	156.000000	0.017800	9.094583
41	160.000000	0.028790	28.454175
42	164.000000	0.061710	33.422376
43	168.000000	0.111200	30.304037
44	172.000000	0.164800	43.239372

Table V.5.6. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	1300.000000	1198.064340
2	1000.000000	1106.591010
3	800.000000	966.103235
4	600.000000	819.332743
5	400.000000	688.923587
6	200.000000	580.115689
7	150.000000	489.838187
8	100.000000	412.102746
9	90.000000	341.389800
10	80.000000	273.657363
11	70.000000	207.444345
12	60.000000	144.181169
13	55.000000	89.285390
14	50.000000	48.247832
15	45.000000	22.360252
16	40.000000	8.612934
17	35.000000	2.551308
18	30.000000	0.442247
19	25.000000	-0.042787
20	20.000000	-0.053278
21	18.000000	-0.012243

Table V.5.6. (Continued)

***** THEORETICAL CALCULATION

	ANGLE	DATA	THEORY
1	0.000000	6118.000000	5048.088970
2	4.000000	5665.000000	4705.075650
3	8.000000	4481.000000	3795.650560
4	12.000000	2992.000000	2619.493660
5	16.000000	1651.000000	1514.483990
6	20.000000	732.199997	714.579119
7	24.000000	269.500000	281.777897
8	28.000000	128.000000	133.345951
9	32.000000	130.299999	127.607557
10	36.000000	150.400000	147.437007
11	40.000000	139.299999	139.059753
12	44.000000	101.900000	102.794300
13	48.000000	60.550000	60.973860
14	52.000000	31.280000	31.161281
15	56.000000	17.230000	17.113929
16	60.000000	13.280000	13.326732
17	64.000000	12.800000	12.856083
18	68.000000	11.740000	11.723842
19	72.000000	9.224000	9.183613
20	76.000000	6.131000	6.124223
21	80.000000	3.578000	3.589190
22	84.000000	2.070000	2.072684
23	88.000000	1.470000	1.466087
24	92.000000	1.346000	1.345410
25	96.000000	1.305000	1.308775
26	100.000000	1.160000	1.160301
27	104.000000	0.900400	0.898703
28	108.000000	0.610100	0.609401
29	112.000000	0.372300	0.372860
30	116.000000	0.226900	0.227087
31	120.000000	0.167200	0.166911
32	124.000000	0.159900	0.159955
33	128.000000	0.168700	0.168979
34	132.000000	0.169500	0.169360
35	136.000000	0.153400	0.153223
36	140.000000	0.123500	0.123514
37	144.000000	0.087250	0.087373
38	148.000000	0.052860	0.052847
39	152.000000	0.027530	0.027501
40	156.000000	0.017800	0.017821
41	160.000000	0.028790	0.028754
42	164.000000	0.061710	0.061725
43	168.000000	0.111200	0.111349
44	172.000000	0.164800	0.164654

Table V.5.6. (Continued)

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	1300.000000	1198.064340	3.954102
2	1000.000000	1106.591010	3.827143
3	800.000000	966.103235	3.609161
4	600.000000	819.332743	3.339596
5	400.000000	688.923587	3.040162
6	200.000000	580.115689	2.720091
7	150.000000	489.838187	2.391558
8	100.000000	412.102746	2.075179
9	90.000000	341.389800	1.790038
10	80.000000	273.657363	1.544042
11	70.000000	207.444345	1.335212
12	60.000000	144.181169	1.156593
13	55.000000	89.285390	0.998984
14	50.000000	48.247832	0.851433
15	45.000000	22.360252	0.703742
16	40.000000	8.612934	0.550553
17	35.000000	2.551308	0.395068
18	30.000000	0.442247	0.248598
19	25.000000	-0.042787	0.128186
20	20.000000	-0.053278	0.047896
21	18.000000	-0.012243	0.009810

	STD. DEV.	CORRELATION																				
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	3.954102D+00	100																				
2	3.827143D+00	100	100																			
3	3.609161D+00	99	100	100																		
4	3.339596D+00	98	99	100	100																	
5	3.040162D+00	95	97	98	100	100																
6	2.720091D+00	93	94	96	98	100	100															
7	2.391558D+00	90	91	94	96	98	99	100														
8	2.075179D+00	86	88	90	93	95	97	99	100													
9	1.790038D+00	81	82	85	87	90	93	96	99	100												
10	1.544042D+00	73	74	77	79	83	86	91	95	99	100											
11	1.335212D+00	63	64	66	69	73	78	83	89	94	98	100										
12	1.156593D+00	50	51	54	58	62	67	73	80	87	94	98	100									
13	9.989845D-01	36	38	41	45	50	56	63	70	78	86	93	98	100								
14	8.514332D-01	23	25	28	33	39	45	52	59	68	76	85	93	98	100							
15	7.037425D-01	12	14	17	22	28	35	42	49	57	66	76	85	93	98	100						
16	5.505533D-01	4	5	9	14	20	26	33	40	48	56	66	76	85	93	98	100					
17	3.950678D-01	-2	-1	2	7	13	19	26	32	39	47	56	66	76	86	93	98	100				
18	2.485983D-01	-6	-4	-2	2	7	13	19	26	32	39	47	57	67	77	86	93	98	100			
19	1.281865D-01	-8	-6	-4	-1	4	9	15	20	26	32	40	48	57	67	77	86	93	98	100		
20	4.789598D-02	-8	-7	-6	-3	1	6	11	16	21	27	33	40	49	58	67	77	86	93	98	100	
21	9.810110D-03	-8	-7	-6	-4	0	4	8	12	17	22	27	33	40	48	57	66	76	85	92	98	100

VI. THE COMPANION CODE LEAST

For the benefit of users more familiar with least squares than with Bayes' method, a companion program LEAST has been developed to facilitate direct comparison of the two methods. LEAST differs from BAYES in two respects only: (1) Updated parameter values are obtained via solution of the least-squares equations (Eqs. (A30) and (A32)) rather than via Bayes' equations. (2) Sequential analyses of two or more data sets are not accepted, since every least-squares analysis assumes that there exists no prior information about the values of the parameters.

LEAST was designed to be totally compatible with BAYES, so that user-supplied subroutines and input for BAYES are correct for LEAST. Because input parameter covariance matrices are not required for least-squares analyses, they need not be provided in LEAST; if provided, they will be ignored.

Data covariance matrices, however, are required for least-squares analyses and must be provided. Off-diagonal elements are permitted in LEAST, unlike many least-squares solvers (e.g. LSFØDF [WH79]; see also [PE80]). Sophisticated searching techniques are not employed in LEAST.

The examples from the previous section are reconsidered here, and those analyses which do not require the sequential feature of Bayes' equations and which do not involve singular data covariance matrices are repeated using least-squares. In all cases the user-supplied FØRTRAN and teletype commands and responses are identical to what is given in Section V, with the obvious substitution of LEAST for BAYES in the teletype commands, and omission of the request for a new data file. Only the output is shown here, in Tables VI.1 through VI.6.

In most cases results from the least-squares analysis agree to within four or more significant digits with results obtained via Bayes' method. This is to be expected, since the input parameter uncertainties were set very large, in which limit Bayes' equations are equivalent to the least-squares equations. The one notable exception is example 3, solution method 2 (Tables V.3.7 and VI.4), for which the agreement is no better than one significant digit. For the BAYES analysis, the prior information indicates that the input uncertainty on parameter number 4 is quite small; repeating the BAYES analysis with this uncertainty increased to 100.0 (i.e., assuming there is no prior information on that parameter) gives results which agree with those obtained from LEAST to three significant digits.

A FØRTRAN listing of the routines unique to LEAST (i.e., not shared by BAYES) is given in Appendix D. The code ZLEAST (not listed here) incorporates dynamic allocation of array storage in the same manner as ZBAYES (see Section IV.6).

Table VI.1. Output from Least-Squares Analysis of Data Given in Example 1. (Compare with Table V.1.4.)

```
SOLVING LEAST SQUARES EQUATIONS.
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000
```

***** LEAST-SQUARES WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL		ENERGY	RESIDUAL
(1)	10.0000	1.68663D+00	(13)	22.0000	5.32793D+00
(2)	11.0000	2.55703D+00	(14)	23.0000	4.97849D+00
(3)	12.0000	3.46260D+00	(15)	24.0000	4.62581D+00
(4)	13.0000	4.32619D+00	(16)	25.0000	4.28129D+00
(5)	14.0000	5.07585D+00	(17)	26.0000	3.95227D+00
(6)	15.0000	5.65895D+00	(18)	27.0000	3.64301D+00
(7)	16.0000	6.04957D+00	(19)	28.0000	3.35564D+00
(8)	17.0000	6.24814D+00	(20)	29.0000	3.09076D+00
(9)	18.0000	6.27507D+00	(21)	30.0000	2.84799D+00
(10)	19.0000	6.16229D+00			
(11)	20.0000	5.94530D+00			
(12)	21.0000	5.65768D+00			

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	1.000000	1.050000
2	-25.000000	-25.200000
3	300.000000	296.000001

***** LEAST-SQUARES WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL		ENERGY	RESIDUAL
(1)	10.0000	-1.05414D-01	(13)	22.0000	-1.61170D-01
(2)	11.0000	-1.31102D-01	(14)	23.0000	-1.48624D-01
(3)	12.0000	-1.55817D-01	(15)	24.0000	-1.36499D-01
(4)	13.0000	-1.77458D-01	(16)	25.0000	-1.25038D-01
(5)	14.0000	-1.94310D-01	(17)	26.0000	-1.14372D-01
(6)	15.0000	-2.05365D-01	(18)	27.0000	-1.04557D-01
(7)	16.0000	-2.10420D-01	(19)	28.0000	-9.55968D-02
(8)	17.0000	-2.09966D-01	(20)	29.0000	-8.74606D-02
(9)	18.0000	-2.04952D-01	(21)	30.0000	-8.00997D-02
(10)	19.0000	-1.96518D-01			
(11)	20.0000	-1.85791D-01			
(12)	21.0000	-1.73749D-01			

Table VI.1. (Continued)

***** INTERMEDIATE RESULTS

	OLD PARAM	NEW PARAMETERS
1	1.000000	1.048810
2	-25.000000	-25.200000
3	300.000000	296.000001

***** LEAST-SQUARES WEIGHTED RESIDUALS AT FORMER VALUES OF PARAMETERS

	ENERGY	RESIDUAL
(1)	10.0000	-5.96750D-05
(2)	11.0000	-7.43780D-05
(3)	12.0000	-8.82044D-05
(4)	13.0000	-1.00781D-04
(5)	14.0000	-1.10312D-04
(6)	15.0000	-1.16280D-04
(7)	16.0000	-1.19279D-04
(8)	17.0000	-1.18910D-04
(9)	18.0000	-1.16275D-04
(10)	19.0000	-1.11483D-04
(11)	20.0000	-1.05250D-04
(12)	21.0000	-9.84983D-05

	ENERGY	RESIDUAL
(13)	22.0000	-9.13200D-05
(14)	23.0000	-8.43045D-05
(15)	24.0000	-7.74271D-05
(16)	25.0000	-7.08651D-05
(17)	26.0000	-6.47221D-05
(18)	27.0000	-5.92667D-05
(19)	28.0000	-5.42286D-05
(20)	29.0000	-4.96142D-05
(21)	30.0000	-4.54124D-05

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	1.000000	1.048809	0.008328
2	-25.000000	-25.200000	0.638696
3	300.000000	296.000001	5.470518

	STD. DEV.	CORRELATION 1	2	3
1	8.327929D-03	100		
2	6.386963D-01	-99	100	
3	5.470518D+00	96	-99	100

Table VI.2. Output from Least-Squares Analysis of Combined Data Set for Example 2. (Compare with Table V.2.7.)

```
SOLVING LEAST SQUARES EQUATIONS.
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
  1      10.000000      10.120099
  2      12.000000      11.525184
  3      17.000000      16.475995

******* NEW VALUES FOR PARAMETERS

          POLD          PNEW        UNCERTAINTY
  1      10.000000     10.123076       0.707745
  2      12.000000     11.525885       0.184360
  3      17.000000     16.479823       3.053504

       STD. DEV.     CORRELATION

                      1    2    3
 1  7.077446D-01  100
 2  1.843599D-01   13  100
 3  3.053504D+00   10   13  100
```

Table VI.3. Output from Least-Squares Analysis of Data Set for Example 3, First Method of Solution. (Compare with Table V.3.4.)

```
SOLVING LEAST SQUARES EQUATIONS.
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS    0.001000

***** INTERMEDIATE RESULTS

         OLD PARAM        NEW PARAMETERS
    1     80.000000       76.810304
    2     50.000000       51.298677
    3     10.000000       12.669581

***** INTERMEDIATE RESULTS

         OLD PARAM        NEW PARAMETERS
    1     80.000000       82.379465
    2     50.000000       51.495605
    3     10.000000       13.817789

***** INTERMEDIATE RESULTS

         OLD PARAM        NEW PARAMETERS
    1     80.000000       83.037786
    2     50.000000       51.481371
    3     10.000000       13.914404

******* NEW VALUES FOR PARAMETERS

            POLD            PNEW        UNCERTAINTY
    1     80.000000       83.046605        3.586875
    2     50.000000       51.480844        0.347024
    3     10.000000       13.920079        0.861652

        STD. DEV.      CORRELATION

                         1    2    3
  1   3.586875D+00  100
  2   3.470243D-01   -2  100
  3   8.616517D-01   18   -2  100
```

Table VI.4. Output from Least-Squares Analysis of Data Set for Example 3, Second Method of Solution. (Compare with Table VI.3.7.)

```
SOLVING LEAST SQUARES EQUATIONS.
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
   1     80.000000      76.817676
   2     50.000000      51.298678
   3     10.000000      12.672309
   4     40.166463      39.967020

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
   1     80.000000      82.843214
   2     50.000000      51.494127
   3     10.000000      14.016209
   4     40.166463      34.514574

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
   1     80.000000      83.997927
   2     50.000000      51.474992
   3     10.000000      14.274148
   4     40.166463      33.101852

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
   1     80.000000      84.081284
   2     50.000000      51.472523
   3     10.000000      14.314622
   4     40.166463      32.948571
```

Table VI.4. (Continued)

******* NEW VALUES FOR PARAMETERS

	POLD	PNEW	UNCERTAINTY
1	80.000000	84.089851	3.829384
2	50.000000	51.472090	0.345052
3	10.000000	14.320542	0.982562
4	40.166463	32.928753	3.653031

	STD. DEV.	CORRELATION 1	2	3	4
1	3.829384D+00	100			
2	3.450518D-01	-3	100		
3	9.825619D-01	34	-2	100	
4	3.653031D+00	-64	4	-88	100

Table VI.5. Output for Least-Squares Analysis of Data Set for Example 4, First Method of Solution. (Compare with Table V.4.4.)

```
SOLVING LEAST SQUARES EQUATIONS.
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
   1    120.000000     123.522848
   2     30.000000      23.859483

***** INTERMEDIATE RESULTS

        OLD PARAM       NEW PARAMETERS
   1    120.000000     123.221544
   2     30.000000      22.601618

******* NEW VALUES FOR PARAMETERS

           POLD           PNEW       UNCERTAINTY
   1    120.000000     123.219435       0.065499
   2     30.000000      22.595013       0.189108

       STD. DEV.     CORRELATION

                        1   2
  1   6.549897D-02  100
  2   1.891082D-01    3 100
```

Table VI.6. Output from Least-Squares Analysis of Data Set for Example 4, Second Method of Solution. (Compare with Table V.4.7.)

```
SOLVING LEAST SQUARES EQUATIONS.
MAXIMUM NUMBER OF ITERATIONS IS  5
CONVERGENCE FACTOR IS   0.001000

***** INTERMEDIATE RESULTS

        OLD PARAM        NEW PARAMETERS
   1    120.000000     123.319824
   2     30.000000      24.335206
   3      1.140000       1.208207

***** INTERMEDIATE RESULTS

        OLD PARAM        NEW PARAMETERS
   1    120.000000     123.216864
   2     30.000000      22.519767
   3      1.140000       1.134418

***** INTERMEDIATE RESULTS

        OLD PARAM        NEW PARAMETERS
   1    120.000000     123.218895
   2     30.000000      22.643222
   3      1.140000       1.137664

******* NEW VALUES FOR PARAMETERS

           POLD          PNEW       UNCERTAINTY
   1    120.000000    123.219183      0.065387
   2     30.000000     22.641993      0.189234
   3      1.140000      1.137609      0.008223

       STD. DEV.     CORRELATION

                       1   2   3
  1  6.538734D-02 100
  2  1.892344D-01   3 100
  3  8.222874D-03  -2  78 100
```

VII. SUMMARY

This technical memorandum was prepared primarily as a guide for the experimentalist who wishes to use Bayes' method as an aid in data analysis. The five examples presented in Sections V and VI were chosen because they typify the kinds of problems encountered in neutron physics studies at ORELA. Nevertheless, the method has far broader applicability; Bayes' method could be used wherever traditional least-squares' methods are now being used, and should be used wherever prior information about parameter values is available. It is the author's hope that this report has offered some insight into the proper use of Bayes' method, and that the reader is inspired to employ Bayes' method in his own research.

REFERENCES

DO79 J. J. Dongarra, C. B. Moler, J. R. Bunch and G. W. Stewart, LINPACK Users' Guide, SIAM (1979).

GA73 A. Gandini, "Nuclear Data and Integral Measurements Correlations for Fast Reactors," Part I and II, Comitato Nazionale Energia Nucleare, Casaccia, Italy (1973), RT/FI(73)5 and 22 (1973).

HE80 D. M. Hetrick and C. Y. Fu, "GLUCS: A Generalized Least-Squares Program for Updating Cross-Section Evaluations with Correlated Data Sets", ORNL/TM-7341, ENDF-303, October 1980.

HE82 D. M. Hetrick, C. Y. Fu and D. C. Larson, to be published.

LA80 N. M. Larson and F. G. Perey, "User's Guide for SAMMY: A Computer Model for Multilevel R-matrix Fits to Neutron Data Using Bayes' Equations", ORNL/TM-7485, ENDF-297, November 1980.

MA80 J. H. Marable, C. R. Weisbin, and G. de Saussure, "Combination of Differential and Integral Data", in "Sensitivity and Uncertainty Analysis of Reactor Performance Parameters", to be published in Adv. Nucl. Sci. and Technology, Plenum Press, editor Martin Becker.

NAG Numerical Algorithms Group (USA) Inc., 1250 Grace Court, Downers Grove, Illinois 60516 - USA. Telephone (312)971-2337.

PE67 F. G. Perey, computer code GENØA, ORNL, unpublished (1967).

PE79 F. G. Perey, lecture notes "Introduction to Probability Theory and Applications to Neutron and Reactor Physics Problems", unpublished.

PE80 R. W. Peelle, "Uncertainty in the Nuclear Data Used for Reactor Calculations", in "Sensitivity and Uncertainty Analysis of Reactor Performance Parameters", to be published in Adv. Nucl. Sci. and Technology, Plenum Press, editor Martin Becker.

PE82 F. G. Perey, submitted to Phys. Rev. A.

RA82 Radiation Shielding Information Center, Oak Ridge National Laboratory, P. O. Box X, Oak Ridge, TN 37830.

WH79 D. M. Wharton and D. K. Olsen, "LSFØDF: A Generalized Nonlinear Least-Squares Fitting Program for Use with ORELA ODF Files", ORNL/TM-6545, December 1978.

APPENDIX A
ALGEBRAIC DETAILS

This appendix presents algebraic details which would obscure the broad understanding of our method, were they to be presented in the text. Included are the derivation of Bayes' equations from Bayes' theorem, development of an appropriate iteration scheme for Bayes' theorem, derivation of least-squares equations from Bayes' equations, proof that a constant covariance matrix is equivalent to a coherent data correction, and derivation of the correct covariance matrix for angular distribution data.

1. Deriving Bayes' Equations

In Section II, we state that Bayes' equations may be derived directly from Bayes' theorem,

$$p(P|DX) \propto p(P|X)p(D|PX), \tag{A1}$$

provided the three basic assumptions are met. These assumptions are:

i) the prior pdf is a joint normal. That is, the pdf for the parameters, prior to consideration of the data D, is

$$p(P|X) \propto \exp\{-1/2\ (P-\bar{P})^t M^{-1}(P-\bar{P})\} \tag{A2}$$

ii) the likelihood function is a joint normal. That is, the pdf for the experimental data is

$$p(D|PX) \propto \exp\{-1/2\ (D-T)^t V^{-1}\ (D-T)\} \tag{A3}$$

where T is a function of the parameter P.

iii) the true value is a linear function of the parameters. That is, a Taylor expansion of the theoretical values around the prior expectation values of the parameters truncates after the linear term:

$$T = \bar{T} + G(P-\bar{P}) \tag{A4}$$

where the sensitivity matrix G is defined by

$$G_{ik} = \left. \frac{\partial T_i}{\partial P_k} \right|_{P=\bar{P}} \tag{A5}$$

and the theoretical value $\bar{T}_i$ (i.e., $\bar{T}$ for data point i) is also evaluated at $P = \bar{P}$.

Given these three assumptions, the posterior pdf $p(P|DX)$ is also a joint normal distribution and may be written

$$p(P|DX) \propto \exp\{-1/2\ (P-\bar{P}')^t\ (M')^{-1}(P-\bar{P}')\}. \tag{A6}$$

Substituting Eq. (A2) through (A6) into Eq. (A1) and equating the exponents yield, in matrix form,

$$\begin{aligned}(P - \bar{P}')^t(M')^{-1}(P - \bar{P}') + Y = {} & (P - \bar{P})^t M^{-1}(P - \bar{P}) \\ & + [D - \bar{T} - G(P - \bar{P})]^t V^{-1}\ [D - \bar{T} - G(P - \bar{P})]\end{aligned} \tag{A7}$$

where Y represents the normalization constant and is independent of P.

Setting $P - \bar{P} = P - \bar{P}' + \bar{P}' - \bar{P}$ in Eq. (A7), and rearranging terms, we obtain

$$\begin{aligned}(P-\bar{P}')^t(M')^{-1}(P-\bar{P}')+Y = {} & (P-\bar{P}')^t(M^{-1}+G^tV^{-1}G)(P-\bar{P}') \\ & + (P-\bar{P}')^t[(M^{-1}+G^tV^{-1}G)(\bar{P}'-\bar{P})-G^tV^{-1}(D-\bar{T})] \\ & + [(\bar{P}'-\bar{P})^t(M^{-1}+G^tV^{-1}G)-(D-\bar{T})^tV^{-1}G](P-\bar{P}') \\ & + (\bar{P}'-\bar{P})^tM^{-1}(\bar{P}'-\bar{P}) \\ & + [D-\bar{T}-G(\bar{P}'-\bar{P})]^tV^{-1}[D-\bar{T}-G(\bar{P}'-\bar{P})]\ .\end{aligned} \tag{A8}$$

Because Eq. (A8) must hold for all values of P, we may equate terms quadratic, linear, or constant in $(P-\bar{P}')$. Equating the quadratic terms gives Bayes' equation for updating the covariance matrix:

$$(M')^{-1} = M^{-1} + G^t\ V^{-1}\ G \tag{A9}$$

Multiplying both sides by M on the left and M' on the right yields

$$M' = [1+MG^tV^{-1}G]^{-1}M \tag{A10}$$

Using the identity $X^{-1} = (ZX)^{-1}Z$ with $Z = G^t(N+V)^{-1}G$, substituting N for GMG^t, and rearranging give

$$M' = [G^t\{(N+V)^{-1}(1+NV^{-1})\}G]^{-1}G^t(N+V)^{-1}GM \ . \tag{A11}$$

The quantity in curly brackets in Eq. (A11) is equal to V^{-1}; making that substitution and introducing the identity $V^{-1}V = I$ into that equation give

$$M' = [G^tV^{-1}G]^{-1}G^tV^{-1}V(N+V)^{-1}GM \ . \tag{A12}$$

Algebraic manipulation then yields

$$M' = [G^tV^{-1}G]^{-1}G^tV^{-1}GM - [G^tV^{-1}G]^{-1}G^tV^{-1}GMG^t(N+V)^{-1}GM \tag{A13}$$

which reduces to

$$M' = M - MG^t(N+V)^{-1}GM \ . \tag{A14}$$

This is exactly Bayes' equation for updating the covariance matrix, Eq. (II.9). Explicitly, this equation may be written

$$M'_{k\ell} = M_{k\ell} - \sum_{n=1}^{K} \sum_{i=1}^{L} \sum_{j=1}^{L} \sum_{m=1}^{K} M_{kn}G_{in}\left((N+V)^{-1}\right)_{ij}G_{jm}M_{m\ell} \tag{A15}$$

where N is given by

$$N_{ij} = \sum_{k=1}^{K} \sum_{\ell=1}^{K} G_{ik}M_{k\ell}G_{j\ell} \ . \tag{A16}$$

To obtain Bayes' equation for updating the parameter values, we equate the linear terms of Eq. (A8). Since the left-hand side of that equation has no terms linear in $(P-\bar{P}')$, the coefficient of $(P-\bar{P}')$ on the right-hand side must be zero. That is,

$$(M^{-1}+G^tV^{-1}G)(\bar{P}'-\bar{P}) = G^tV^{-1}(D-\bar{T}) \quad . \tag{A17}$$

From Eq. (A9), the first quantity on the left is just $(M')^{-1}$; we therefore multiply both sides of Eq. (A17) by M', using Eq. (A14), and obtain

$$\bar{P}'-\bar{P} = (M-MG^t(N+V)^{-1}GM)G^tV^{-1}(D-\bar{T}) \tag{A18}$$

which reduces to

$$\bar{P}'-\bar{P} = MG^t(N+V)^{-1}(D-\bar{T}) \quad . \tag{A19}$$

Explicitly, this equation is

$$\bar{P}'_k = \bar{P}_k + \sum_{\ell=1}^{K} \sum_{i=1}^{L} \sum_{j=1}^{L} M_{k\ell}G_{i\ell}\left((N+V)^{-1}\right)_{ij}(D_j-\bar{T}_j) \tag{A20}$$

Finally, we note that the constant term in Eq. (A8) may be specified using Eq. (A19) to give

$$Y = (D-\bar{T})^t[(N+V)^{-1}GMG^t(N+V)^{-1} + (1-(N+V)^{-1}N)V^{-1}(1-N(N+V)^{-1})](D-\bar{T}) \tag{A21}$$

which reduces to

$$Y = (D-\bar{T})^t(N+V)^{-1}(D-\bar{T}) \tag{A22}$$

2. Iteration Scheme

The linearity hypothesis, i.e., the assumption that the Taylor expansion of the theoretical values around the prior expectation value truncates after the linear term, is in general only approximately true. Therefore, the parameter values $\bar{P}'$ resulting from application of Bayes' equations are also only approximately correct. To obtain more accurate values, the Taylor expansion, Eq. (A4), may be performed not around $\bar{P}$ but around the new (intermediate) values $\bar{P}^{(n)}$, where n represents the nth iteration and $\bar{P}^{(0)} = \bar{P}$:

$$T \simeq \bar{T}^{(n)} + G^{(n)}(P-\bar{P}^{(n)}) \tag{A23}$$

Here the sensitivity matrix $G^{(n)}$ and the theoretical values $\bar{T}^{(n)}$ are evaluated at $P = \bar{P}^{(n)}$. With Eq. (A23) for T, the formula analogous to Eq. (A7) is

$$(P-\bar{P}^{(n+1)})^t(M^{(n+1)})^{-1}(P-\bar{P}^{(n+1)}) + Y = (P-\bar{P})^t M^{-1}(P-\bar{P})$$

$$+[D-\bar{T}^{(n)}-G^{(n)}(P-\bar{P}^{(n)})]^t V^{-1}[D-\bar{T}^{(n)}-G^{(n)}(P-\bar{P}^{(n)})] \tag{A24}$$

Setting P equal to $P - \bar{P}^{(n+1)} + \bar{P}^{(n+1)}$ everywhere in the right-hand side of Eq. (A24) gives the formula analogous to Eq. (A8) with $\bar{T}$ in that expression replaced by $\bar{T}^{(n)}+G^{(n)}(\bar{P}-\bar{P}^{(n)})$, and G by $G^{(n)}$. The iterative form of Bayes' equations follow immediately:

$$\bar{P}^{(n+1)} = \bar{P} + MG^{(n)t}\ (N^{(n)}+V)^{-1}\ (D-\bar{T}^{(n)}-G^{(n)}(\bar{P}-\bar{P}^{(n)})) \tag{A25}$$

$$M^{(n+1)} = M - MG^{(n)t}\ (N^{(n)} + V)^{-1}\ G^{(n)}M \tag{A26}$$

where

$$N^{(n)} = G^{(n)}M\ G^{(n)t} \tag{A27}$$

3. Derivation of Least-Squares from Bayes' Equations

The equivalence of the least-squares method with Bayes' equations in the limit of large M (i.e., in the extreme case where there is no prior knowledge of the values of the parameters) is best demonstrated by considering Eq. (A17):

$$(M^{-1}+G^{t}V^{-1}G)(\bar{P}'-\bar{P}) = G^{t}V^{-1}(D-\bar{T}) \tag{A28}$$

or, in the iterative form,

$$(M^{-1} + G^{(n)t}V^{-1}\, G^{(n)})\ (\bar{P}^{(n+1)}-\bar{P}) \tag{A29}$$

$$= G^{(n)t}\ V^{-1}\ (D-\bar{T}^{(n)} - G^{(n)}\ (\bar{P}-\bar{P}^{(n)}))$$

For large M, the term $G^{t}V^{-1}G$ overwhelms M^{-1}, and this equation reduces immediately to

$$\bar{P}^{(n+1)} = \bar{P}^{(n)} + (G^{(n)t}\ V^{-1}G^{(n)})^{-1}\ G^{(n)t}V^{-1}\ (D-\bar{T}^{(n)}) \tag{A30}$$

which is the well-known least-squares formula.

Similarly, parameter uncertainties and covariances can be found from the iterative form of Eq. (A9):

$$(M^{(n+1)})^{-1} = M^{-1} + G^{(n)t}\ V^{-1}\ G^{(n)} \tag{A31}$$

which reduces to

$$M^{(n+1)} = (G^{(n)t}V^{-1}G^{(n)})^{-1} \tag{A32}$$

in the limit of large M. In least-squares applications, the quoted covariance matrix is usually the value given in Eq. (A32), scaled by χ^2/d, where

$$\chi^2 = (D-\bar{T}^{(n)})^t V^{-1} (D-\bar{T}^{(n)}), \tag{A33}$$

and d is the number of degrees of freedom in the problem (d = number of data points minus number of parameters). In the LEAST program described in this manual, no such scaling is performed.

4. Use of Data Covariances to Indicate Coherent Data Corrections

In Example V.3 we saw computational proof that a constant term added to the data covariance matrix is equivalent to a constant, coherent correction to the theory. For a mathematical proof, we begin by writing the covariance matrix in the form

$$V = \bar{V} + AA^t \tag{A34}$$

where $\bar{V}$ is the original covariance matrix, and A is a column matrix whose elements are zero outside the range where the correction is to be applied, and constant inside the range. Bayes' equations require the inverse of $N + \bar{V} + AA^t$, which is equivalent to

$$(N+\bar{V}+AA^t)^{-1} = (N+\bar{V})^{-1} \left[1 - \frac{1}{1 + A^t (N+\bar{V})^{-1} A} AA^t (N+\bar{V})^{-1}\right] \tag{A35}$$

as can be verified by multiplying the right-hand side by $(N + \bar{V} + AA^t)$. Substitution of this expression into the first of Bayes' equations (Eq. (A19) or Eq. (II.8)) gives

$$\bar{P}' = \bar{P} + MG^t(N+\bar{V})^{-1} (D- \Delta T-\bar{T}) \tag{A36}$$

where the correction term ΔT is given by

$$\Delta T = \frac{1}{1 + A^t \ (N+\bar{V})^{-1} A} AA^t \ (N+\bar{V})^{-1} \ (D-\bar{T}) \tag{A37}$$

Thus, the presence of an additive constant term (AA^t) in the data covariance matrix is equivalent to a coherent correction to the theory or, equivalently, to the data. Also, note that if the elements of column matrix A are not zero or constant, Eqs. (A36) and (A37) remain valid, but the correction term ΔT varies from point to point.

5. Covariance Matrix for Differential Cross Sections

In Example V.5, Eq. (V.5.6) was stated to be the correct form for the covariance matrix for angular distribution data. Here we present a proof that this form is in fact correct.

Let p be the probability that a particle is scattered to within $d\Omega$ of position (θ,ϕ). (Azimuthal symmetry is assumed.) Then p is given by

$$p = \frac{d\sigma/d\Omega \ d \ (\cos \theta) \ d\phi}{\sigma} \tag{A38}$$

Discretizing gives the probability for finding the particle at θ_i as

$$p_i = \frac{2\pi D_i \sin\theta_i \Delta\theta_i}{\sigma} \tag{A39}$$

Perey [PE79] has shown that the covariance matrix for the p_i is given by

$$<\delta p_i \ \delta p_j> = \frac{p_i \ \delta_{ij} - p_i p_j}{N} \tag{A40}$$

where N may be viewed as a normalization constant related to the duration of the experiment. Using Eq. (A39) to replace both δp_i and p_i in Eq. (A40) gives

$$\frac{2\pi}{\sigma} <\delta D_i \ \delta D_j> \sin\theta_j \ \sin\theta_i \ \Delta\theta_i \ \Delta\theta_j$$

$$= \frac{2\pi}{\sigma N} D_i \ \sin\theta_i \Delta\theta_i \delta_{ij} - \left(\frac{2\pi}{\sigma}\right)^2 \frac{1}{N} D_i \ \sin\theta_i \Delta\theta_i \ D_j \ \sin\theta_j \Delta\theta_j \tag{A41}$$

Therefore the covariance between data D_i and data D_j is $<\delta D_i \ \delta D_j>$, or

$$V_{ij} = \frac{D_i}{N \sin\theta_i \Delta\theta_i} \delta_{ij} - \frac{\dfrac{D_i \ \sin\theta_i \Delta\theta_i}{N \sin\theta_i \Delta\theta_i} \ \dfrac{D_j \ \sin\theta_j \Delta\theta_j}{N \sin\theta_j \Delta\theta_j}}{\dfrac{\sigma}{2\pi N}} \tag{A42}$$

Using Eq. (V.5.3), $\frac{\sigma}{2\pi N}$ can be written as

$$\frac{\sigma}{2\pi N} = \sum_i \frac{D_i}{N \sin\theta_i \Delta\theta_i} \quad (\sin \theta_i \ \Delta\theta_i)^2 \tag{A43}$$

If we now identify

$$\frac{D_i}{N\ \sin\theta_i \Delta\theta_i}$$

as the square of the uncertainty with which D_i is measured, Eqs. (A42) and (A43) become

$$V_{ij} = (\Delta D_i)^2 \delta_{ij} - \frac{(\Delta D_i)^2 \sin\theta_i \Delta\theta_i (\Delta D_j)^2 \sin\theta_j \Delta\theta_j}{\sum_k (\Delta D_k\ \sin\theta_k \Delta\theta_k)^2} \tag{A44}$$

which is identical to Eq. (V.5.6).

APPENDIX B
NOTATION USED IN BAYES

NOTATION USED IN BAYES

FØRTRAN Name (Dimensions, if Applicable)	Symbol Used in Text	Meaning
AUTØ		flag which, when equal to 'Y', indicates that the automatic numerical derivative option is in effect
CØNVER		convergence fraction; if no parameter changes more than this fraction of its former value, iteration ceases
DATA(NDATMX)	D	experimental data points
DATCØV(NDATMX,NDATMX)	V	covariance matrix for the data
DUM(NDATMX)		dummy array for temporary storage
E(NDATMX)		first independent variable (often, energy)
E2(NDATMX)		second independent variable
EMG(NDATMX,NPARMX)	$MG^{(n)t}$	
EN(m), where m= NDATMX*(NDATMX + 1))/2	$N^{(n)}+V$	
G(NDATMX,NPARMX)	$G^{(n)}$	$G_{ij}^{(n)}$ = partial derivative of T with respect to parameter j, evaluated at data point i and parameter values $P^{(n)}$
HBASE		fractional step size for generating numerical derivatives automatically
IDUM(NPARMX)		dummy array
IFPAR(NNPARM)		flag which is set to 1 if parameter is to be varied, 0 if not

FØRTRAN Name (Dimensions, if Applicable)	Symbol Used in Text	Meaning
ITMAX		maximum number of iterations allowed
ITER		current iteration number
IWHERE(NNPARM)		bookkeeper
IYDUM(NNPARM)		dummy array
NDAT	L	number of experimental data points
NDATMX		array dimensions for data
NNPARM		array dimensions for parameters (complete set, varied and fixed)
NPARAM		total number of parameters (both varied and fixed)
NPAR	K	number of varied parameters for the problem
NPARMX		array dimensions for varied parameters
PARAM(NNPARM)		initial estimate for the complete set of parameters
PARCØV(NNPARM,NNPARM)		initial estimate for the covariance matrix for the complete set of parameters
PARM(NPARMX)	$\bar{P}^{(n)}$	current values of varied parameters
PARM(NPARMX)	$\bar{P}^{(n+1)}$	updated values of varied parameters

FØRTRAN Name (Dimensions, if Applicable)	Symbol Used in Text	Meaning
PARVAR(NPARMX,NPARMX)	M	temporary storage for parameter covariance matrix
PDUM(NPARMX)		dummy array for temporary storage
PØLD(NPARMX)	$\overline{P}$	initial estimate for value of varied parameters
RCØND		condition number determined by the matrix solution routine DPPCØ; if very small, matrix is nearly singular
SIG(NDATMX)		reciprocal of the square-root of the diagonal element of EN, used in scaling of matrix manipulations
TH(NDATMX)	$\overline{T}^{(n)}$	theoretical values for data points, evaluated at parameter values $\overline{P}$
VARDAT(m),where m=(NDATMX*(NDATMX+1))/2	V	data covariance matrix in crunched form
VARPAR(m),where m=(NPARMX*(NPARMX+1))/2	M	parameter covariance matrix in crunched form
VARNEW(m),where m=(NPARMX*(NPARMX+1))/2	$M^{(n+1)}$	updated parameter covariance matrix

APPENDIX C
FØRTRAN LISTING OF BAYES MAIN PROGRAM AND OTHER ROUTINES

```
C
C
C
      PROGRAM  BAYES
C
C
C     PROGRAM TO SOLVE BAYES' EQUATIONS FOR UPDATING
C             PARAMETER VALUES BASED ON INPUT DATA AND
C             APPROPRIATE THEORY (TO BE PROVIDED BY
C             THE USER)
C
C     N.M.LARSON, MARCH 1981
C
C ************************************************************
C
C     INPUT TO BAYES PROGRAM IS ON UNIT 5(TELETYPE) -- THE PROGRAM
C       WILL PROMPT YOU BY ASKING APPROPRIATE QUESTIONS.
C
C     OUTPUT FROM BAYES IS ON UNIT 6.
C
C ************************************************************
C
C     FOR BAYES METHOD :  SOLVE THESE EQUATIONS
C         (P BAR SUPER N+1) - (P BAR)
C             = M * (G TRANSPOSE) *
C                   ((N+V) INVERSE) *
C                   ( D - (T BAR SUPER N)
C                        - G * (P BAR - P BAR SUPER N) )
C     AND
C         M - (M SUPER N+1) = M * (G TRANPOSE)
C                               * ((N+V) INVERSE) * G * M
C
C     WHERE
C         N = G * M * (G TRANPOSE)
C
C
C     NOTATION IS AS FOLLOWS --
C
C     FORTRAN  THEORETICAL          MEANING
C      NAME       NAME
C     -------  -----------    ------------------------------
C     E                       ENERGY, OR FIRST INDEPENDENT
C                               VARIABLE
C     E2                      SECOND INDEPENDENT VARIABLE
C     NDAT                    NUMBER OF DATA POINTS
C     NPAR                    NUMBER OF PARAMETERS TO BE VARIED
```

```
C       DATA    D                    EXPERIMENTAL DATA
C       DATCOV  V                    DATA COVARIANCE MATRIX
C       VARDAT  V                    COVARIANCE MATRIX FOR DATA, STORED
C                                      IN TRIANGULAR FORM AS
C                                           1 2 4 7 ... OR  1
C                                             3 5 8         2 3
C                                               6 9         4 5 6
C                                                10         7 8 9 10
C                                                           ...
C       POLD    P BAR                INITIAL ESTIMATE OF PARAMETERS
C       PARVAR  M                    ESTIMATE OF COVARIANCE MATRIX FOR
C                                      PARAMETERS, STORED IN REGULAR
C                                      MATRIX FORM
C       VARPAR  M                    ESTIMATE OF COVARIANCE MATRIX FOR
C                                      PARAMETERS, STORED IN TRIANGULAR FORM
C       TH      T BAR SUPER N        THEORETICAL VALUE FOR DATA, EVALUATED
C                                      AT CURRENT VALUE OF PARAMETERS
C       G       G                    PARTIAL DERIV OF (TH) WITH
C                                      RESPECT TO PARAMETERS, EVALUATED
C                                      AT ENERGY E AND AT CURRENT VALUE
C                                      OF PARAMETERS, STORED AS G(NDAT,NPAR)
C       PARM    P BAR SUPER N        INTERMEDIATE (OR CURRENT) VALUES OF
C                                      PARAMETERS
C       PARM    P BAR SUPER N+1 NEW VALUES OF PARAMETERS
C       VARNEW  M SUPER N+1          NEW VALUES FOR PARAMETER COVARIANCES
C
C ***************************************************************
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL JJPAR, JJDAT, JJTH, JJG, JJREL, JJCHL,
     *     JJREB, JJCHB, JJGMG
C
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
C *** 325=(25*(25+1))/2
C
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
C *** 1326=(51*(51+1))/2
C
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      DATA NDATMX /51/, NPARMX /25/, NNPARM/26/
      DATA YES /1HY/
```

```
C
C
      CALL OUTALL(JJPAR, JJDAT, JJTH, JJG, JJREL, JJCHL,
     *     JJREB, JJCHB, JJGMG)
C               OUTALL READS FILE 'BAYES.OUT' TO DETERMINE
C               WHICH OUTPUT IS TO BE SUPPRESSED
C
      WRITE (5,99999)
      READ (5,99998) ITMAX, CONVER
      WRITE (6,99997) ITMAX, CONVER
C
      WRITE (5,99996)
      READ (5,99995) AUTO
      IF (AUTO.NE.YES) GO TO 10
C
      WRITE (5,99994)
      READ (5,99993) HBASE
      WRITE (6,99992) HBASE
C
C
   10 CALL PPARAM(NNPARM)
C               OUTPUT FROM PPARAM -- NPAR,PARM,VARPAR,
C                                     NPARAM,PARAM,PARCOV,POLD
C
      IF (NPAR.GT.NPARMX) GO TO 40
C
      IF (JJPAR) CALL OUTPAR
C               INPUT TO OUTPAR -- PARM,VARPAR
C               OUTPAR WRITES OUT PARAMETERS, COVARIANCE MATRIX
C
      IF (JJPAR .AND. NPAR.LT.NPARAM) CALL OUTYPR
C               INPUT TO OUTYPR -- PARAM,PARCOV,IFPAR
C               OUTYPR WRITES OUT COMPLETE LIST OF PARAMETERS
C                       AND COVARIANCE MATRIX
C
   20 CALL SETDAT
C               OUTPUT FROM SETDAT -- NDAT,E,DATA,VARDAT
C
      CALL FIXV
C               INPUT TO FIXV -- DATCOV (STORED IN VARDAT AND EN)
C               OUTPUT FROM FIXV -- VARDAT
C
      IF (NDAT.GT.NDATMX) GO TO 50
C
      IF (JJDAT) CALL OUTDAT
C               INPUT TO OUTDAT -- NDAT,E,DATA,VARDAT
C               OUTDAT WRITES OUT DATA, COVARIANCE MATRIX
C
      ITER = 0
```

```
C
   30 CALL THEORY(AUTO, HBASE)
C               THEORY GENERATES THE FUNCTION AND ITS DERIVATIVES
C               INPUT TO THEORY -- NPAR,NDAT,PARM,E
C               OUTPUT FROM THEORY -- TH,G
C
      IF (JJTH) CALL OUTTH
C               INPUT TO OUTTH -- NDAT,E,DATA,TH
C               OUTTH PRINTS THEORY (TH) AND DATA (DATA) VS ENERGY (E)
C
      IF (JJG) CALL OUTG
C               INPUT TO OUTG -- NPAR,NDAT,E,G
C               OUTG PRINTS PARTIAL DERIVATIVES AT PARM
C
C
      CALL NEWPAR(JJREL, JJCHL, JJREB, JJCHB, JJGMG)
C               INPUT TO NEWPAR -- POLD,PARM,VARPAR,DATA,VARDAT,TH
C               OUTPUT FROM NEWPAR -- PARM (UPDATED), VARNEW
C               ALSO USED BY NEWPAR -- EMG,EN,DUM,SIG,PDUM
C
      IF (ITER.LT.ITMAX) CALL OUTP1
C               INPUT TO OUTP1 -- POLD,PARM
      IF (ITER.EQ.ITMAX) CALL OUTP2
C               INPUT TO OUTP2 -- POLD,PARM,VARNEW
C
C
      CALL UPDATE
C               INPUT TO UPDATE -- PARM,VARPAR,IWHERE
C               OUTPUT FROM UPDATE -- PARAM,PARCOV, UPDATED ALSO
      IF (ITER.EQ.ITMAX .AND. NPAR.LT.NPARAM) CALL OUTYPR
C               INPUT TO OUTYPR -- PARAM,PARCOV,IFPAR
C
C
      ITER = ITER + 1
      IF (ITER.LE.ITMAX) GO TO 30
C
C
      CALL RESTRT
      GO TO 20
C               FOR MORE DATA TO BE INCORPORATED
C
C
C
   40 WRITE (5,99991) NPAR, NPARMX
      STOP
C
   50 WRITE (5,99990) NDAT, NDATMX
      STOP
```

```
C
99999 FORMAT (42H HOW MANY ITERATIONS?  WHAT IS CONVERGENCE,
     *       11H FRACTION? $)
99998 FORMAT (I, F)
99997 FORMAT (32H MAXIMUM NUMBER OF ITERATIONS IS, I3/7H CONVER,
     *       16HGENCE FACTOR IS , F10.6)
99996 FORMAT (39H DO YOU WISH TO USE AUTOMATIC NUMERICAL,
     *       14H DERIVATIVES? $)
99995 FORMAT (A1)
99994 FORMAT (42H WHAT IS FRACTION DIFFERENCE FOR AUTOMATIC,
     *       13H DERIVATIVE? $)
99993 FORMAT (F)
99992 FORMAT (36H AUTOMATIC DERIVATIVE USES STEP SIZE, F10.5)
99991 FORMAT (9H YOU WANT, I5, 21H BUT ARE ALLOWED ONLY, I5,
     *       17HPARAMETER VALUES./28H CHANGE IN COMMONS /PAR/ AND,
     *       8H /BOTH/,, 35H AND IN SUBROUTINES PPARAM AND OLDP)
99990 FORMAT (9H YOU WANT, I5, 21H BUT ARE ALLOWED ONLY, I5,
     *       13H DATA VALUES./30H  CHANGE IN COMMONS /DAT/ AND ,
     *       42H/BOTH/, AND IN SUBROUTINES SETDAT AND FIXV)
      END
C
C
C
      SUBROUTINE PPARAM(NNPARM)
C
C *** PURPOSE -- INITIALIZE ARRAYS PARM AND PARVAR (INITIAL
C ***            VALUES FOR PARAMETERS AND COVARIANCE MATRIX
C ***            FOR PARAMETERS, RESPECTIVELY)
C ***
C *** THIS VERSION ASSUMES A LONG LIST OF PARAMETERS, ONLY SOME
C ***           OF WHICH ARE TO BE VARIED, IS STORED IN PARAM,
C ***           THE COVARIANCE MATRIX OF THE WHOLE LIST IS STORED
C ***           IN PARCOV, AND THE MATRIX ELEMENTS OF IFPAR ARE
C ***           ZERO IF THAT PARAMETER IS NOT TO BE VARIED,
C ***           AND ONE IF IT IS TO BE VARIED.  THESE ARRAYS ARE
C ***           FOUND IN COMMON/YRPAR/, ALONG WITH THE ARRAY
C ***           IWHERE WHICH HANDLES BOOKKEEPING, AND VARIABLE
C ***           NPARAM WHICH IS THE TOTAL NUMBER OF PARAMETERS
C ***           IN THE LIST.
C
C
C       NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      DIMENSION PARVAR(25,25)
      EQUIVALENCE (PARVAR(1,1),VARPAR(1))
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
```

```
C
C
C *** CALL SETPAR TO INITIALIZE ARRAYS PARAM,PARCOV,IFPAR -- I.E.,
C ***   TO INTIALIZE PARAMETERS
      CALL SETPAR
      IF (NPARAM.GT.NNPARM) GO TO 150
C
C
C *** CHECK IF ALL PARAMETERS ARE UNFLAGGED -- IN WHICH CASE,
C ***   FLAG THEM ALL.
      N = 0
      DO 10 I=1,NPARAM
           IF (IFPAR(I).EQ.0) N = N + 1
   10 CONTINUE
      IF (N.LT.NPARAM) GO TO 30
      IF (N.EQ.0) GO TO 80
      DO 20 I=1,NPARAM
           IFPAR(I) = 1
   20 CONTINUE
      GO TO 80
C
C
C *** CHECK WHETHER THE UNFLAGGED PARAMETERS ARE CORRELATED
C ***   TO THE FLAGGED ONES.
   30 DO 60 K=1,NPARAM
           KK = 0
           DO 50 I=1,NPARAM
                IF (IFPAR(I).NE.0) GO TO 50
C *** HERE I ISN'T VARIED, SO CHECK IF IT'S CORRELATED TO A
C ***   PARAMETER THAT IS
                DO 40 J=1,NPARAM
                     IF (J.EQ.I) GO TO 40
                     IF (PARCOV(J,I).EQ.0.D0) GO TO 40
                     IF (IFPAR(J).EQ.0) GO TO 40
C *** HERE J IS VARIED SO I MUST BE TOO
                     IFPAR(I) = 2
                     KK = 1
                     GO TO 50
   40           CONTINUE
   50      CONTINUE
           IF (KK.EQ.0) GO TO 70
   60 CONTINUE
   70 CONTINUE
C
C
C *** COUNT THE NUMBER OF VARIED PARAMETERS ADDED TO THE ORIGINAL
```

```
C ***   LIST.  ALSO INITIALIZE PARM AND IWHERE.
   80 IPAR = 0
      JPAR = 0
      DO 90 I=1,NPARAM
           IF (IFPAR(I).EQ.0) GO TO 90
           IPAR = IPAR + 1
           PARM(IPAR) = PARAM(I)
           IWHERE(IPAR) = I
           IF (IFPAR(I).EQ.2) JPAR = JPAR + 1
   90 CONTINUE
C
      IF (JPAR.NE.0) WRITE (6,99999) JPAR
C
C *** ORGANIZE COVARIANCE MATRIX
      NPAR = IPAR
      IPAR = 0
      DO 110 I=1,NPARAM
           IF (IFPAR(I).EQ.0) GO TO 110
           IPAR = IPAR + 1
           JPAR = 0
           DO 100 J=1,NPARAM
                IF (IFPAR(J).EQ.0) GO TO 100
                JPAR = JPAR + 1
                PARVAR(JPAR,IPAR) = PARCOV(J,I)
  100      CONTINUE
  110 CONTINUE
C
C *** INITIALIZE POLD
      DO 120 I=1,NPAR
           POLD(I) = PARM(I)
  120 CONTINUE
C
C *** REARRANGE STORAGE OF VARPAR TO USE ONLY LOWER TRIANGULAR
      IL = 0
      DO 140 I=1,NPAR
           DO 130 L=1,I
                IL = IL + 1
                VARPAR(IL) = PARVAR(L,I)
  130      CONTINUE
  140 CONTINUE
      RETURN
C
  150 WRITE (5,99998) NPARAM
      WRITE (6,99998) NPARAM
      STOP
C
99999 FORMAT (46HONUMBER OF ADDITIONAL PARAMETERS WHICH MUST BE,
     *     1H , 38HVARIED BECAUSE OF COVARIANCE MATRIX IS, I5)
99998 FORMAT (9H YOU WANT, I5, 21H BUT ARE ALLOWED ONLY, I5,
     *     17HPARAMETER VALUES./28H CHANGE EVERY ARRAY IN COMMO,
     *     1HN, 8H/YRPAR/.)
      END
```

```
C
C
C
      SUBROUTINE FIXV
C
C *** PURPOSE -- REARRANGE DATCOV TO BE VARDAT
C ***            I.E., KEEP ONLY LOWER TRIANGULAR PORTION
C ***            OF DATA COVARIANCE MATRIX
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
C
      IL = 0
      DO 20 I=1,NDAT
           DO 10 L=1,I
                IL = IL + 1
                VARDAT(IL) = DATCOV(L,I)
   10      CONTINUE
   20 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE THEORY(AUTO, HBASE)
C
C *** PURPOSE -- GENERATE TH(IDAT) = THEORY AND G(IDAT,IPAR) =
C ***            PARTIAL DERIVATIVE OF TH(IDAT) WITH RESPECT TO
C ***            PARAMETER PARM(IPAR), EVALUATED AT PARM
C
C *** NOTE -- THERE ARE TWO OPTIONS FOR PARTIAL DERIVATIVES --
C ***           (1) IF AUTO='Y' (IN MAIN PROGRAM) NUMERICAL PARTIAL
C ***               DERIVATIVES ARE GENERATED AUTOMATICALLY BY DO4NML,
C ***               AND THE USER NEED NOT SUPPLY A WORKING VERSION
C ***               OF SUBROUTINE DERIV.
C ***           (2) IF AUTO .NE. 'Y', PARTIAL DERIVATIVES ARE TO
C ***               BE GENERATED IN SUBROUTINE DERIV WHICH IS
C ***               SUPPLIED BY THE USER.
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      COMMON /DERIVA/ KDAT, NDERIV
```

```
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DIMENSION DER(14), EREST(14)
      EXTERNAL FUN
      DATA YES /1HY/
C
C
      IF (HBASE.EQ.0.D0) HBASE = 0.001D0
C                         THIS SHOULD BE TRIED AT SEVERAL VALUES
C
      NDER = 1
      IFAIL = 0
C
      DO 50 IDAT=1,NDAT
           KDAT = IDAT
           TH(IDAT) = THEO(KDAT)
C
           IF (AUTO.EQ.YES) GO TO 20
C ***      HERE THE USER GENERATES DERIVATIVES EXPLICITLY
           IPAR = 0
           DO 10 I=1,NPARAM
                II=I
                IF (IFPAR(I).EQ.0) GO TO 10
                IPAR = IPAR + 1
                G(KDAT,IPAR) = DERIV(KDAT,II)
   10           CONTINUE
           GO TO 50
C
   20      CONTINUE
C ***      HERE SUBROUTINE D04NML IS USED TO OBTAIN
C ***           NUMERICAL DERIVATIVES
           DO 40 IPAR=1,NPAR
                NDERIV = IPAR
                PIPAR = PARM(IPAR)
                CALL D04NML(PIPAR, NDER, HBASE, DER, EREST,
     *               FUN, IFAIL)
                G(IDAT,IPAR) = DER(1)
                IF (EREST(1).LT.0.D0) GO TO 70
                IF (IFAIL.NE.0) GO TO 60
   30           PARM(IPAR) = PIPAR
                PARAM(IWHERE(NDERIV)) = PIPAR
C               CUZ PARM IS CHANGED IN D04NML (EXPLICITLY, IN FUN)
   40      CONTINUE
C
   50 CONTINUE
      RETURN
C
C
   60 WRITE (6,99999)IFAIL
C     STOP
      GO TO 30
```

```
C
   70 WRITE (6,99998) EREST(1)
C     STOP
      GO TO 30
C
99999 FORMAT (30H0******ERROR IN D04NML, IFAIL=,I2)
99998 FORMAT (30H0******ERROR IN D04NML, EREST=,1PG14.6)
      END
C
C
C
      DOUBLE PRECISION FUNCTION FUN(X)
C
C *** PURPOSE -- FUN IS REQUIRED BY AUTOMATIC NUMERICAL DERIVATIVE
C ***            ROUTINE D04NML.  FUN GENERATES THE THEORY AT
C ***            PARM(NDERIV)=X, FOR THE KDAT-TH DATA POINT.
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      COMMON /DERIVA/ KDAT, NDERIV
      PARAM(IWHERE(NDERIV)) = X
      PARM(NDERIV) = X
      FUN = THEO(KDAT)
      RETURN
      END
C
C
C
      SUBROUTINE NEWPAR(JJREL, JJCHL, JJREB, JJCHB, JJGMG)
C
C *** PURPOSE -- SET UP AND SOLVE BAYES' EQUATIONS
C ***
C *** INPUT   -- POLD,PARM,VARPAR,DATA,VARDAT,TH,G
C *** OUTPUT  -- PARM,VARNEW
C *** ALSO USED- EMG,EN,DUM,PDUM,SIG
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL JJREL, JJCHL, JJREB, JJCHB, JJGMG
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
```

```
C
C
C *** TH = DATA - TH - G * (POLD-PARM)
      DO 10 I=1,NDAT
           TH(I) = DATA(I) - TH(I)
   10 CONTINUE
C
C *** CHECK IF LEAST-SQUARES RESIDUALS OR CHI SQUARED ARE WANTED
      IF ( (.NOT.JJREL) .AND. (.NOT.JJCHL) ) GO TO 80
C
C *** INITIALIZE EN
      IJ = 0
      DO 30 I=1,NDAT
           DO 20 J=1,I
                IJ = IJ + 1
                EN(IJ) = VARDAT(IJ)
   20      CONTINUE
   30 CONTINUE
C
C *** SCALE EN
      CALL SCALE(EN, SIG, NDAT, IFDIAG)
C
C *** FACTORIZE EN
      IF (IFDIAG.EQ.0) GO TO 40
      IIIII = 0
      CALL DPPCO(EN, NDAT, RCOND, DUM, IIIII)
      IF (1.D0+RCOND.EQ.1.D0) GO TO 310
   40 CONTINUE
C
C *** GENERATE RESIDUAL V INVERSE * (DATA-THEORY)
      DO 50 I=1,NDAT
           DUM(I) = TH(I)*SIG(I)
   50 CONTINUE
      IF (IDIAG.EQ.1) CALL DPPSL(EN, NDAT, DUM)
      DO 60 I=1,NDAT
           DUM(I) = DUM(I)*SIG(I)
   60 CONTINUE
      IF (JJREL) CALL OUTREL
      IF (.NOT.JJCHL) GO TO 80
      CHI = 0.
      DO 70 I=1,NDAT
           CHI = CHI + TH(I)*DUM(I)
   70 CONTINUE
      CALL OUTCHL(CHI)
   80 CONTINUE
C
      IF (ITER.EQ.0) GO TO 110
      DO 100 IPAR=1,NPAR
           DO 90 I=1,NDAT
                TH(I) = TH(I) - G(I,IPAR)*(POLD(IPAR)-PARM(IPAR))
```

```
   90       CONTINUE
  100 CONTINUE
C
  110 CONTINUE
C
C *** EMG=VARPAR*G
      CALL MUL
C
C *** EN=G*EMG+VARDAT
      CALL MUL2
      IF (JJGMG) CALL OUTGMG(EN, NDAT)
C
C *** SCALE EN
      CALL SCALE(EN, SIG, NDAT, IFDIAG)
      IF (JJGMG) CALL OUTGMG(EN,NDAT)
C
      IF (IFDIAG.EQ.0) GO TO 120
C *** FACTORIZE EN
      IIIII = 0
      CALL DPPCO(EN, NDAT, RCOND, DUM, IIIII)
      IF (1.0D0+RCOND.EQ.1.0D0) GO TO 300
  120 CONTINUE
C
C *** SCALE TH
      DO 130 I=1,NDAT
           DUM(I) = TH(I)*SIG(I)
  130 CONTINUE
C
C *** CALCULATE EN**-1 * DUM
      IF (IFDIAG.EQ.1) CALL DPPSL(EN, NDAT, DUM)
C
C *** RESCALE DUM
      DO 140 I=1,NDAT
           DUM(I) = DUM(I)*SIG(I)
  140 CONTINUE
C
C *** CHECK IF DIAGNOSTICS ARE WANTED
      IF (JJREB) CALL OUTREB(DUM)
      IF (.NOT.JJCHB) GO TO 160
      CHI = 0.
      DO 150 I=1,NDAT
           CHI = CHI + TH(I)*DUM(I)
  150 CONTINUE
      CALL OUTCHB(CHI)
  160 CONTINUE
C
C *** CALCULATE  M * G * EN**-1 * TH
      DO 170 I=1,NPAR
           PDUM(I) = 0.D0
  170 CONTINUE
      DO 190 I=1.NPAR
```

```
          DO 180 J=1,NDAT
               PDUM(I) = PDUM(I) + EMG(J,I)*DUM(J)
  180     CONTINUE
  190 CONTINUE
C
C *** PARM=PDUM+POLD
      ICONV = 0
      DO 210 I=1,NPAR
          A = PDUM(I) + POLD(I)
          IF (CONVER.EQ.0.D0) GO TO 200
          B = ABS((A-PARM(I))/PARM(I))
          IF (B.LE.CONVER) ICONV = ICONV + 1
  200     PARM(I) = A
  210 CONTINUE
      IF (ICONV.EQ.NPAR) ITER = ITMAX
C ***          IE, IF EACH PARAMETER HAS CONVERGED TO WITHIN
C ***                  THE REQUIRED TOLERANCE, DO NO MORE
C ***                  ITERATIONS
C
C
      IF (ITER.NE.ITMAX) GO TO 290
C
C *** INITIALIZE VARNEW
      IJ = 0
      DO 230 I=1,NPAR
          DO 220 J=1,I
               IJ = IJ + 1
               VARNEW(IJ) = VARPAR(IJ)
  220     CONTINUE
  230 CONTINUE
C
      IJ = 0
      DO 280 I=1,NPAR
C
C *** SET AND SCALE DUM
          DO 240 J=1,NDAT
               DUM(J) = EMG(J,I)*SIG(J)
  240     CONTINUE
C
C *** CALCULATE EN**-1 * DUM
          IF (IFDIAG.EQ.1) CALL DPPSL(EN, NDAT, DUM)
C
C *** RESCALE DUM
          DO 250 IDAT=1,NDAT
               DUM(IDAT) = DUM(IDAT)*SIG(IDAT)
  250     CONTINUE
C
C *** UPDATE VARNEW
          DO 270 J=1,I
               IJ = IJ + 1
               DO 260 K=1,NDAT
                    VARNEW(IJ) = VARNEW(IJ) - EMG(K,J)*DUM(K)
```

```
  260           CONTINUE
  270       CONTINUE
  280 CONTINUE
C
  290 RETURN
C
C
  300 WRITE (5,99999) RCOND
      WRITE (6,99999) RCOND
CX      STOP
      GO TO 120
C
  310 WRITE (5,99998) RCOND
      WRITE (6,99998) RCOND
CX      STOP
      GO TO 230
C
99999 FORMAT (10X, 20H********************, /7H RCOND=,
     *     1PD12.6, /24H THE MATRIX IS POSSIBLY , 10HSINGULAR -,
     *     30H- RESULTS MAY NOT BE ACCURATE., /10X, 9H*********,
     *     11H***********)
99998 FORMAT (10X, 20H********************, /7H RCOND=,
     *     1PD12.6, /20H VARDAT IS POSSIBLY , 14HSINGULAR -- LE,
     *     31HAST-SQUARES CHI SQUARED MAY NOT, 13H BE ACCURATE.,
     *     /10X, 20H********************)
      END
C
C
C
      SUBROUTINE SCALE(A, SIG, N, IFDIAG)
C
C *** PURPOSE -- SCALE ARRAY A TO INSURE AGAINST ARTIFICIAL
C ***            SINGULARITIES WHEN SOLVING THE MATRIX EQUATION.
C ***
C *** INPUT   -- A
C *** OUTPUT  -- MODIFIED A, SIG
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(1), SIG(N)
C
      IL = 0
      DO 20 I=1,N
           IL = IL + I
           SI = A(IL)
           IF (SI.LE.0.D0) GO TO 10
           SIG(I) = 1.D0/SQRT(SI)
           GO TO 20
   10      SIG(I) = 1.D0/SQRT(-SI)
   20 CONTINUE
```

```
C
      IFDIAG = 0
      IL = 0
      DO 40 I=1,N
           SI = SIG(I)
           DO 30 L=1,I
                IL = IL + 1
                IF (I.NE.L .AND. A(IL).NE.0.DO) IFDIAG = 1
                A(IL) = A(IL)*SI*SIG(L)
   30      CONTINUE
   40 CONTINUE
C
      RETURN
      END
C
C
C
      SUBROUTINE MUL
C
C *** PURPOSE -- GENERATE EMG(IDAT,IPAR) = (SUM OVER JPAR)
C ***                                      G(IDAT,JPAR) *
C ***                                      VARPAR(JPAR AND IPAR)
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
C
      JK = 0
      DO 60 J=1,NPAR
           DO 10 L=1,NDAT
                EMG(L,J) = 0.DO
   10      CONTINUE
           DO 50 K=1,NPAR
                IF (J.GE.K) GO TO 20
                KJ = (K*(K-1))/2 + J
                EM = VARPAR(KJ)
                GO TO 30
   20           JK = JK + 1
                EM = VARPAR(JK)
   30           CONTINUE
                IF (EM.EQ.0.DO) GO TO 50
                DO 40 L=1,NDAT
                     EMG(L,J) = EMG(L,J) + EM*G(L,K)
   40           CONTINUE
   50      CONTINUE
   60 CONTINUE
      RETURN
      END
```

```
C
C
C
      SUBROUTINE MUL2
C
C *** PURPOSE -- GENERATE EN(IDAT AND JDAT) = VARDAT(IDAT AND JDAT) +
C ***               (SUM OVER IPAR) G(IDAT,IPAR) * EMG(JDAT,IPAR)
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      IL = 0
      DO 20 I=1,NDAT
           DO 10 L=1,I
                IL = IL + 1
                EN(IL) = VARDAT(IL)
   10      CONTINUE
   20 CONTINUE
C
      DO 50 J=1,NPAR
           IL = 0
           DO 40 I=1,NDAT
                DO 30 L=1,I
                     IL = IL + 1
                     EN(IL) = G(I,J)*EMG(L,J) + EN(IL)
   30           CONTINUE
   40      CONTINUE
   50 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE UPDATE
C
C *** PURPOSE -- UPDATE THE ARRAYS PARAM AND PARCOV TO AGREE WITH
C ***            NEW VALUES GENERATED IN NEWPAR, AND STORED IN
C ***            PARM AND VARNEW
C
C NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
```

```
C
      DO 10 I=1,NPAR
           PARAM(IWHERE(I)) = PARM(I)
   10 CONTINUE
C
      IF (ITER.LT.ITMAX) RETURN
C
      IJ = 0
      DO 30 I=1,NPAR
           IW = IWHERE(I)
           DO 20 J=1,I
                IJ = IJ + 1
                PARCOV(IWHERE(J),IW) = VARNEW(IJ)
   20      CONTINUE
   30 CONTINUE
C
      RETURN
      END
C
C
C
      SUBROUTINE RESTRT
C
C *** PURPOSE -- REINITIALIZE POLD AND VARPAR TO MAKE READY
C ***            FOR A NEW DATA SET
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      DO 10 I=1,NPAR
           POLD(I) = PARM(I)
   10 CONTINUE
      IJ = 0
      DO 30 I=1,NPAR
           DO 20 J=1,I
                IJ = IJ + 1
                VARPAR(IJ) = VARNEW(IJ)
   20      CONTINUE
   30 CONTINUE
      RETURN
      END
```

APPENDIX D

FØRTRAN LISTING OF ROUTINES UNIQUE TO PROGRAM LEAST

```
C
C
C
      PROGRAM  LEAST
C
C
C     PROGRAM TO SOLVE LEAST SQUARES EQUATIONS FOR UPDATING
C             PARAMETER VALUES BASED ON INPUT DATA AND
C             APPROPRIATE THEORY (TO BE PROVIDED BY
C             THE USER)
C
C     N.M.LARSON, JANUARY 1981
C
C ***************************************************************
C
C     INPUT TO LEAST PROGRAM IS ON UNIT 5(TELETYPE) -- THE PROGRAM
C       WILL PROMPT YOU BY ASKING APPROPRIATE QUESTIONS.
C
C     OUTPUT FROM LEAST IS ON UNIT 6.
C
C ***************************************************************
C
C     FOR LEAST METHOD :  SOLVE THESE EQUATIONS
C         (P BAR SUPER N+1) - (P BAR SUPER N)
C             = (G TRANSPOSE * V INVERSE * G) INVERSE *
C                   G TRANSPOSE * V INVERSE *
C                   ( D - (T BAR SUPER N)
C     AND
C          (M SUPER N+1) = (G TRANSPOSE * V INVERSE * G) INVERSE
C
C
C     NOTATION IS AS FOLLOWS --
C
C     FORTRAN  THEORETICAL          MEANING
C      NAME       NAME
C     -------  -----------    ------------------------------
C     E                       ENERGY, OR FIRST INDEPENDENT
C                               PARAMETER
C     E2                      SECOND INDEPENDENT PARAMETER
C     NDAT                    NUMBER OF DATA POINTS
C     NPAR                    NUMBER OF PARAMETERS TO BE VARIED
C     DATA     D              EXPERIMENTAL DATA
C     DATCOV   V              DATA COVARIANCE MATRIX
```

```
C       VARDAT  V                 COVARIANCE MATRIX FOR DATA, STORED
C                                   IN TRIANGULAR FORM AS
C                                         1 2 4 7 ... OR  1
C                                           3 5 8         2 3
C                                             6 9         4 5 6
C                                              10         7 8 9 10
C                                                         ...
C       POLD    P BAR             INITIAL ESTIMATE OF PARAMETERS
C       TH      T BAR SUPER N     THEORETICAL VALUE FOR DATA, EVALUATED
C                                   AT CURRENT VALUE OF PARAMETERS
C       G       G                 PARTIAL DERIV OF (TH) WITH
C                                   RESPECT TO PARAMETERS, EVALUATED
C                                   AT ENERGY E AND AT CURRENT VALUE
C                                   OF PARAMETERS, STORED AS G(NDAT,NPAR)
C       PARM    P BAR SUPER N     INTERMEDIATE (OR CURRENT) VALUES OF
C                                   PARAMETERS
C       PARM    P BAR SUPER N+1 NEW VALUES OF PARAMETERS
C       VARNEW  M SUPER N+1     NEW VALUES FOR PARAMETER COVARIANCES
C
C ***************************************************************
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL JJPAR, JJDAT, JJTH, JJG, JJREL, JJCHL,
     *     JJREB, JJCHB, JJGMG
C
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
C *** 66=(11*(11+1))/2
C
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
C *** 1326=(51*(51+1))/2
C
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      DATA NDATMX /51/, NPARMX /25/, NNPARM/ 26/
      DATA YES /1HY/
C
C
      CALL OUTALL(JJPAR, JJDAT, JJTH, JJG, JJREL, JJCHL,
     *     JJREB, JJCHB, JJGMG)
```

```
C
      WRITE (5,99999)
      WRITE (6,99999)
      WRITE (5,99998)
      READ (5,99997) , ITMAX, CONVER
      WRITE (6,99996) ITMAX, CONVER
C
      WRITE (5,99995)
      READ (5,99994) AUTO
      IF (AUTO.NE.YES) GO TO 10
C
      WRITE (5,99993)
      READ (5,99992) HBASE
      WRITE (6,99991) HBASE
C
C
   10 CALL PPARAM(NNPARM)
C               OUTPUT FROM PPARAM -- NPAR,PARM,PARAM,POLD,NPARAM
C
      IF (NPAR.GT.NPARMX) GO TO 30
C
      CALL SETDAT
C               OUTPUT FROM SETDAT -- NDAT,E,DATA,VARDAT
C
      CALL FIXV
C               INPUT TO FIXV -- DATCOV (STORED IN VARDAT AND EN)
C               OUTPUT FROM FIXV -- VARDAT
C
      IF (NDAT.GT.NDATMX) GO TO 40
C
      IF (JJDAT) CALL OUTDAT
C               INPUT TO OUTDAT -- NDAT,E,DATA,VARDAT
C               OUTDAT WRITES OUT DATA, COVARIANCE MATRIX
C
C
      ITER = 0
   20 CALL THEORY(AUTO, HBASE)
C               THEORY GENERATES THE FUNCTION AND ITS DERIVATIVES
C               INPUT TO THEORY -- NPAR,NDAT,PARM,E
C               OUTPUT FROM THEORY -- TH,G
C
      IF (JJTH) CALL OUTTH
C               INPUT TO OUTTH -- NDAT,E,DATA,TH
C               OUTTH PRINTS THEORY (TH) AND DATA (DATA) VS ENERGY (E)
C
      IF (JJG) CALL OUTG
C               INPUT TO OUTG -- NPAR,NDAT,E,G
C               OUTG PRINTS PARTIAL DERIVATIVES AT PARM
```

```
C
C
      CALL NEWPRL(JJREL, JJCHL)
C               INPUT TO NEWPRL -- POLD,PARM,DATA,VARDAT,TH
C               OUTPUT FROM NEWPRL -- PARM (UPDATED), VARNEW
C               ALSO USED BY NEWPRL -- EMG,EN,DUM,SIG,PDUM,VARPAR
C
      IF (ITER.LT.ITMAX) CALL OUTP1
C               INPUT TO OUTP1 -- POLD,PNEW
      IF (ITER.EQ.ITMAX) CALL OUTP2
C               INPUT TO OUTP2 -- POLD,PNEW,VARNEW
C
C
      CALL UPDATE
C               INPUT TO UPDATE -- PARM,VARPAR,IWHERE
C               OUTPUT FROM UPDATE -- PARAM,PARCOV, UPDATED ALSO
      IF (ITER.EQ.ITMAX .AND. NPAR.LT.NPARAM) CALL OUTYPR
C               INPUT TO OUTYPR -- PARAM,PARCOV,IFPAR
C
C
      ITER = ITER + 1
      IF (ITER.LE.ITMAX) GO TO 20
C
C
      STOP
C
C
C
   30 WRITE (5,99990) NPAR, NPARMX
      STOP
C
   40 WRITE (5,99989) NDAT, NDATMX
      STOP
C
99999 FORMAT (33H SOLVING LEAST SQUARES EQUATIONS.)
99998 FORMAT (42H HOW MANY ITERATIONS?  WHAT IS CONVERGENCE,
     *     11H FRACTION? $)
99997 FORMAT (I, F)
99996 FORMAT (32H MAXIMUM NUMBER OF ITERATIONS IS, I3/7H CONVER,
     *     16HGENCE FACTOR IS , F10.6)
99995 FORMAT (39H DO YOU WISH TO USE AUTOMATIC NUMERICAL,
     *     14H DERIVATIVES? $)
99994 FORMAT (A1)
99993 FORMAT (42H WHAT IS FRACTION DIFFERENCE FOR AUTOMATIC,
     *     13H DERIVATIVE? $)
99992 FORMAT (F)
99991 FORMAT (36H AUTOMATIC DERIVATIVE USES STEP SIZE, F10.5)
```

```
99990 FORMAT (9H YOU WANT, I5, 21H BUT ARE ALLOWED ONLY, I5,
     *      17HPARAMETER VALUES./28H CHANGE IN COMMONS /PAR/ AND,
     *      8H /BOTH/,, 35H AND IN SUBROUTINES PPARAM AND OLDP)
99989 FORMAT (9H YOU WANT, I5, 21H BUT ARE ALLOWED ONLY, I5,
     *      13H DATA VALUES./30H  CHANGE IN COMMONS /DAT/ AND ,
     *      42H/BOTH/, AND IN SUBROUTINES SETDAT AND FIXV)
      END
C
C
C
      SUBROUTINE PPARAM(NNPARM)
C
C *** PURPOSE -- INITIALIZE ARRAYS PARM AND IFPAR
C ***
C *** THIS VERSION ASSUMES A LONG LIST OF PARAMETERS (ONLY SOME
C ***           OF WHICH ARE TO BE VARIED) IS STORED IN PARAM.
C ***           THE MATRIX ELEMNTS OF IFPAR ARE
C ***           ZERO IF THAT PARAMETER IS NOT TO BE VARIED,
C ***           AND ONE IF IT IS TO BE VARIED.  THESE ARRAYS ARE
C ***           FOUND IN COMMON/YRPAR/, ALONG WITH THE ARRAY
C ***           IWHERE WHICH HANDLES BOOKKEEPING, AND VARIABLE
C ***           NPARAM WHICH IS THE TOTAL NUMBER OF PARAMETERS
C ***           IN THE LIST.
C
C
C       NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      DIMENSION PARVAR(25,25)
      EQUIVALENCE (PARVAR(1,1),VARPAR(1))
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
C
C *** CALL SETPAR TO INITIALIZE ARRAYS PARAM,PARCOV,IFPAR -- I.E.,
C ***   TO INTIALIZE PARAMETERS
      CALL SETPAR
      IF (NPARAM.GT.NNPARM) GO TO 60
C
C
C *** CHECK IF ALL PARAMETERS ARE UNFLAGGED -- IN WHICH CASE,
C ***   FLAG THEM ALL.
      N = 0
      DO 10 I=1,NPARAM
          IF (IFPAR(I).EQ.0) N = N + 1
```

```
   10 CONTINUE
      IF (N.LT.NPARAM) GO TO 30
      DO 20 I=1,NPARAM
           IFPAR(I) = 1
   20 CONTINUE
C
C
C *** INITIALIZE PARM AND IWHERE.
   30 IPAR = 0
      DO 40 I=1,NPARAM
           IF (IFPAR(I).EQ.0) GO TO 40
           IPAR = IPAR + 1
           PARM(IPAR) = PARAM(I)
           IWHERE(IPAR) = I
   40 CONTINUE
      NPAR = IPAR
C
C *** INITIALIZE POLD
      DO 50 I=1,NPAR
           POLD(I)=PARM(I)
   50 CONTINUE
      RETURN
C
   60 WRITE (5,99999) NPARAM
      WRITE (6,99999) NPARAM
      STOP
C
99999 FORMAT (9H YOU WANT, I5, 21H BUT ARE ALLOWED ONLY, I5,
     *     17HPARAMETER VALUES./28H CHANGE EVERY ARRAY IN COMMO,
     *     1HN, 8H/YRPAR/.)
      END
C
C
C
      SUBROUTINE FIXV
C
C *** PURPOSE -- REARRANGE DATCOV TO BE VARDAT
C ***            I.E., KEEP ONLY LOWER TRIANGULAR PORTION
C ***            OF DATA COVARIANCE MATRIX
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      DIMENSION DATCOV(51,51)
      EQUIVALENCE (DATCOV(1,1),VARDAT(1))
C
      IL = 0
      DO 20 I=1,NDAT
           DO 10 L=1,I
                IL = IL + 1
                VARDAT(IL) = DATCOV(L,I)
```

```
   10       CONTINUE
   20 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE THEORY(AUTO, HBASE)
C
C *** PURPOSE -- GENERATE TH(IDAT) = THEORY AND G(IDAT,IPAR) =
C ***            PARTIAL DERIVATIVE OF TH(IDAT) WITH RESPECT TO
C ***            PARAMETER PARM(IPAR), EVALUATED AT PARM
C
C *** NOTE -- THERE ARE TWO OPTIONS FOR PARTIAL DERIVATIVES --
C ***           (1) IF AUTO='Y' (IN MAIN PROGRAM) NUMERICAL PARTIAL
C ***               DERIVATIVES ARE GENERATED AUTOMATICALLY BY D04NML,
C ***               AND THE USER NEED NOT SUPPLY A WORKING VERSION
C ***               OF SUBROUTINE DERIV.
C ***           (2) IF AUTO .NE. 'Y', PARTIAL DERIVATIVES ARE TO
C ***               BE GENERATED IN SUBROUTINE DERIV WHICH IS
C ***               SUPPLIED BY THE USER.
C
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      COMMON /DERIVA/ KDAT, NDERIV
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      DIMENSION DER(14), EREST(14)
      EXTERNAL FUN
      DATA YES /1HY/
C
C
      IF (HBASE.EQ.0.D0) HBASE = 0.001D0
C                     THIS SHOULD BE TRIED AT SEVERAL VALUES
C
      NDER = 1
      IFAIL = 0
C
      DO 50 IDAT=1,NDAT
           KDAT = IDAT
           TH(IDAT) = THEO(KDAT)
C
           IF (AUTO.EQ.YES) GO TO 20
```

```
C ***      HERE THE USER GENERATES DERIVATIVES EXPLICITLY
           IPAR = 0
           DO 10 I=1,NPARAM
                II=I
                IF (IWHERE(I).EQ.0) GO TO 10
                IPAR = IPAR + 1
                G(KDAT,IPAR) = DERIV(KDAT,II)
   10           CONTINUE
           GO TO 50
C
   20      CONTINUE
C ***      HERE SUBROUTINE D04NML IS USED TO OBTAIN
C ***           NUMERICAL DERIVATIVES
           DO 40 IPAR=1,NPAR
                NDERIV = IPAR
                PIPAR = PARM(IPAR)
                CALL D04NML(PIPAR, NDER, HBASE, DER, EREST,
     *               FUN, IFAIL)
                G(IDAT,IPAR) = DER(1)
                IF (EREST(1).LT.0.D0) GO TO 60
                IF (IFAIL.NE.0) GO TO 60
   30           PARM(IPAR) = PIPAR
                PARAM(IWHERE(NDERIV)) = PIPAR
C               CUZ PARM IS CHANGED IN D04NML (EXPLICITLY, IN FUN)
   40      CONTINUE
C
   50 CONTINUE
      RETURN
C
C
   60 WRITE (6,99999)
C     STOP
      GO TO 30
C
99999 FORMAT (22H0******ERROR IN D04NML)
      END
C
C
C
      DOUBLE PRECISION FUNCTION FUN(X)
C
C *** PURPOSE -- FUN IS REQUIRED BY AUTOMATIC NUMERICAL DERIVATIVE
C ***            ROUTINE D04NML.  FUN GENERATES THE THEORY AT
C ***            PARM(NDERIV)=X, FOR THE KDAT-TH DATA POINT.
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
```

```
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
      COMMON /DERIVA/ KDAT, NDERIV
      PARAM(IWHERE(NDERIV)) = X
      PARM(NDERIV) = X
      FUN = THEO(KDAT)
      RETURN
      END
C
C
C
      SUBROUTINE NEWPRL(JJREL, JJCHL)
C
C *** PURPOSE -- SET UP AND SOLVE LEAST SQUARES EQUATIONS
C ***
C *** INPUT   -- POLD,PARM,VARPAR,DATA,VARDAT,TH,G
C *** OUTPUT  -- PARM,VARNEW
C *** ALSO USED- EMG,EN,DUM,PDUM,SIG
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL JJREL, JJCHL
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DIMENSION PSIG(1)
C       DIMENSION PSIG(NPAR)
      EQUIVALENCE (PSIG(1),G(1,1))
C
C
C *** TH = DATA - TH
      DO 10 I=1,NDAT
           TH(I) = DATA(I) - TH(I)
   10 CONTINUE
C
C *** INITIALIZE EN
      IJ = 0
      DO 30 I=1,NDAT
           DO 20 J=1,I
                IJ = IJ + 1
                EN(IJ) = VARDAT(IJ)
   20      CONTINUE
   30 CONTINUE
C
C *** SCALE EN
      CALL SCALE(EN, SIG, NDAT, IFDIAG)
C
```

```
C *** FACTORIZE EN
      IF (IFDIAG.EQ.0) GO TO 40
      IIIII = 0
      CALL DPPCO(EN, NDAT, RCOND, DUM, IIIII)
      IF (1.D0+RCOND.EQ.1.D0) GO TO 280
   40 CONTINUE
C
C *** CHECK IF DIAGNOSTICS ARE WANTED
      IF ( (.NOT.JJREL) .AND. (.NOT.JJCHL) ) GO TO 80
C
C *** GENERATE RESIDUAL V INVERSE * (DATA-THEORY)
      DO 50 I=1,NDAT
           DUM(I) = TH(I)*SIG(I)
   50 CONTINUE
      IF (IDIAG.EQ.1) CALL DPPSL(EN, NDAT, DUM)
      DO 60 I=1,NDAT
           DUM(I) = DUM(I)*SIG(I)
   60 CONTINUE
      IF (JJREL) CALL OUTREL
      IF (.NOT.LCHL) GO TO 80
      CHI = 0.
      DO 70 I=1,NDAT
           CHI = CHI + TH(I)*DUM(I)
   70 CONTINUE
      CALL OUTCHL(CHI)
   80 CONTINUE
C
C
C *** GENERATE EMG = G TRANSPOSE * V INVERSE
      DO 110 I=1,NPAR
           DO 90 J=1,NDAT
                DUM(J) = G(J,I)*SIG(J)
   90      CONTINUE
C
           IF (IFDIAG.EQ.1) CALL DPPSL(EN, NDAT, DUM)
C
           DO 100 J=1,NDAT
                EMG(J,I) = DUM(J)*SIG(J)
  100      CONTINUE
  110 CONTINUE
C
C *** GENERATE M SUPER (N+1) INVERSE = G TRANSPOSE * V INVERSE * G
      IJ = 0
      DO 130 I=1,NPAR
           DO 120 J=1,I
                IJ = IJ + 1
                VARPAR(IJ) = 0.D0
  120      CONTINUE
  130 CONTINUE
```

```
C
      IJ = 0
      DO 160 I=1,NPAR
           DO 150 J=1,I
                IJ = IJ + 1
                DO 140 K=1,NDAT
                     VARPAR(IJ) = VARPAR(IJ) + EMG(K,J)*G(K,I)
  140           CONTINUE
  150      CONTINUE
  160 CONTINUE
C
C *** SCALE VARPAR
      CALL SCALE(VARPAR, PSIG, NPAR, IFDIAG)
C
C *** FACTORIZE VARPAR
      IF (IFDIAG.EQ.0) GO TO 170
  165 IIIII = 0
      CALL DPPCO(VARPAR, NPAR, RCOND, PDUM, IIIII)
      IF (1.D0+RCOND.EQ.1.D0) GO TO 290
  170 CONTINUE
C
C *** SET PDUM = G TRANSPOSE * V INVERSE * (DATA - T BAR)
      DO 180 I=1,NPAR
           PDUM(I) = 0.D0
  180 CONTINUE
      DO 200 I=1,NPAR
           DO 190 J=1,NDAT
                PDUM(I) = PDUM(I) + EMG(J,I)*TH(J)
  190      CONTINUE
  200 CONTINUE
C
C *** SCALE PDUM
      DO 210 I=1,NPAR
           PDUM(I) = PDUM(I)*PSIG(I)
  210 CONTINUE
C
C *** GENERATE VARPAR INVERSE TIMES PDUM
      IF (IFDIAG.EQ.1) CALL DPPSL(VARPAR, NPAR, PDUM)
C
C *** RESCALE PDUM, CHECK CONVERGENCE, ADD PARM
      ICONV = 0
      DO 230 I=1,NPAR
           A = PDUM(I)*PSIG(I)
           IF (CONVER.EQ.0.D0) GO TO 220
           B = DABS(A/PARM(I))
           IF (B.LE.CONVER) ICONV = ICONV + 1
  220      CONTINUE
C          IF (DABS(A).LT.DABS(PARM(I))*0.3D0) GO TO 225
C               TYPE 99,ITER,I
C99             FORMAT(' ITER,IPAR=',2I3)
C          A=A*0.5
C          GO TO 220
```

```
  225       PARM(I) = PARM(I) + A
  230 CONTINUE
      IF (ICONV.EQ.NPAR) ITER = ITMAX
C ***           IE, IF EACH PARAMETER HAS CONVERGED TO WITHIN
C ***             THE REQUIRED TOLERANCE, DO NO MORE ITERATIONS
C
C
      IF (ITER.NE.ITMAX) GO TO 270
C
C *** GENERATE VARNEW = (G TRANSPOSE * V INVERSE * G) INVERSE
      IJ = 0
      DO 260 IPAR=1,NPAR
           DO 240 I=1,NPAR
                PDUM(I) = 0.D0
  240      CONTINUE
           PDUM(IPAR) = 1.D0*PSIG(IPAR)
           IF (IFDIAG.EQ.1) CALL DPPSL(VARPAR, NPAR, PDUM)
           DO 250 J=1,IPAR
                IJ = IJ + 1
                VARNEW(IJ) = PDUM(J)*PSIG(J)
  250      CONTINUE
  260 CONTINUE
C
  270 RETURN
C
  280 WRITE (5,99999) RCOND
      WRITE (6,99999) RCOND
      GO TO 40
C
  290 WRITE (5,99998) RCOND
      WRITE (6,99998) RCOND
      GO TO 170
C
99999 FORMAT (10X, 20H********************, /7H RCOND=,
     *     1PD12.6, /34H VARDAT IS POSSIBLY SINGULAR -- RE,
     *     26HSULTS MAY NOT BE ACCURATE., /10X, 12H************,
     *     8H********)
99998 FORMAT (10X, 20H********************, /7H RCOND=,
     *     1PD12.6, /34H GT*V-1*G IS POSSIBLY SINGULAR -- ,
     *     28HRESULTS MAY NOT BE ACCURATE., /10X, 10H**********,
     *     10H**********)
      END
C
C
C
      SUBROUTINE SCALE(A, SIG, N, IFDIAG)
C
C *** PURPOSE -- SCALE ARRAY A TO INSURE AGAINST ARTIFICIAL
C ***            SINGULARITIES WHEN SOLVING THE MATRIX EQUATION.
C ***
C *** INPUT   -- A
C *** OUTPUT  -- MODIFIED A, SIG
```

```
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(1), SIG(N)
C
      IL = 0
      DO 20 I=1,N
           IL = IL + I
           SI = A(IL)
           IF (SI.LE.0.D0) GO TO 10
           SIG(I) = 1.D0/SQRT(SI)
           GO TO 20
   10      SIG(I) = 1.D0/SQRT(-SI)
   20 CONTINUE
C
      IFDIAG = 0
      IL = 0
      DO 40 I=1,N
           SI = SIG(I)
           DO 30 L=1,I
                IL = IL + 1
                IF (I.NE.L .AND. A(IL).NE.0.D0) IFDIAG = 1
                A(IL) = A(IL)*SI*SIG(L)
   30      CONTINUE
   40 CONTINUE
C
      RETURN
      END
C
C
C
      SUBROUTINE UPDATE
C
C *** PURPOSE -- UPDATE THE ARRAYS PARAM AND PARCOV TO AGREE WITH
C ***            NEW VALUES GENERATED IN NEWPAR, AND STORED IN
C ***            PARM AND VARNEW
C
C NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      DO 10 I=1,NPAR
           PARAM(IWHERE(I)) = PARM(I)
   10 CONTINUE
```

```
C
      IF (ITER.LT.ITMAX) RETURN
C
      IJ = 0
      DO 30 I=1,NPAR
           IW = IWHERE(I)
           DO 20 J=1,I
                IJ = IJ + 1
                PARCOV(IWHERE(J),IW) = VARNEW(IJ)
   20      CONTINUE
   30 CONTINUE
C
      RETURN
      END
```

APPENDIX E
FØRTRAN LISTING OF OUTPUT ROUTINES

```
C
C
C
      SUBROUTINE OUTALL(JJPAR, JJDAT, JJTH, JJG,
     *     JJREL, JJCHL, JJREB, JJCHB, JJGMG)
C
C       PURPOSE -- READ FILE BAYES.OUT AND SET LOGICAL FLAGS
C                  TO DECIDE WHAT OUTPUT IS TO BE SUPPRESSED
C
      LOGICAL JJPAR, JJDAT, JJTH, JJG, JJREL, JJCHL,
     *     JJREB, JJCHB, JJGMG, JJTRUE, JJFALS
      DIMENSION BB(14)
C
      DATA JJTRUE/.TRUE./, JJFALS/.FALSE./
      DATA BLANK/'   '/, OUT/'OUT'/
      DATA BB/'PAR', 'DAT', 'RES', 'CHI', 'TH ', 'G  ',
     *     'REL', 'CHL', 'REB', 'CHB', 'GMG', 'TH       ',
     *     'G  ', 'G           '/
C
      JJPAR = JJTRUE
      JJDAT = JJTRUE
      JJTH = JJTRUE
      JJG = JJTRUE
      JJREL = JJTRUE
      JJCHL = JJTRUE
      JJREB = JJTRUE
      JJCHB = JJTRUE
      JJGMG = JJTRUE
C
      OPEN (UNIT=11,FILE='BAYES.OUT')
   10 CONTINUE
      READ (11,10000,END=40,ERR=40) A,B
10000 FORMAT(2A3)
      IF (A.NE.OUT .AND. A.NE.BLANK) WRITE (5,10100) A,B
10100 FORMAT(42H FILE BAYES.OUT CONTAINS THIS UNRECOGNIZED,
     *     12H LINE ***** ,2A3,6H *****)
      IF (A.NE.OUT) GO TO 10
      DO 20 I=1,14
        IF (B.EQ.BB(I)) GO TO 30
   20 CONTINUE
      WRITE (5,10100) A,B
      GO TO 10
```

```
   30 CONTINUE
      IF (I.EQ.1) JJPAR = JJFALS
      IF (I.EQ.2) JJDAT = JJFALS
      IF (I.EQ.3) JJREL = JJFALS
      IF (I.EQ.3) JJREB = JJFALS
      IF (I.EQ.4) JJCHL = JJFALS
      IF (I.EQ.4) JJCHB = JJFALS
      IF (I.EQ.5) JJTH = JJFALS
      IF (I.EQ.6) JJG = JJFALS
      IF (I.EQ.7) JJREL = JJFALS
      IF (I.EQ.8) JJCHL = JJFALS
      IF (I.EQ.9) JJREB = JJFALS
      IF (I.EQ.10) JJCHB = JJFALS
      IF (I.EQ.11) JJGMG = JJFALS
      IF (I.EQ.12) JJTH = JJFALS
      IF (I.EQ.13) JJG = JJFALS
      IF (I.EQ.14) JJG = JJFALS
      GO TO 10
   40 RETURN
      END
C
C
C
      SUBROUTINE OUTPAR
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      WRITE (6,99999)
C
      II = 0
      DO 10 I=1,NPAR
           II = II + I
           PDUM(I) = DSQRT(VARPAR(II))
   10 CONTINUE
C
      WRITE (6,99998)
      WRITE (6,99997) (I,PARM(I),PDUM(I),I=1,NPAR)
C
      II = 0
      IOFF = 0
      IALL = 0
      DO 30 I=1,NPAR
           DO 20 J=1,I
                II = II + 1
                IF (VARPAR(II).NE.0.D0) IALL = IALL + 1
                IF (I.EQ.J) GO TO 30
                IF (VARPAR(II).NE.0.D0) IOFF = IOFF + 1
```

```
   20        CONTINUE
   30   CONTINUE
        IF (IOFF.NE.0) CALL OUTV(VARPAR, PDUM, IDUM, NPAR)
        IF (IALL.NE.0) RETURN
        WRITE (6,99996)
        STOP
C
99999 FORMAT (29H0******INPUT PARAMETER VALUES)
99998 FORMAT (36H0          PARAMETER      UNCERTAINTY)
99997 FORMAT (I5, 2F14.6)
99996 FORMAT (46H******ERROR.  COVARIANCE MATRIX FOR PARAMETERS,
     *     1H , 41HBE INITIALIZED IN SUBROUTINE PARAM.******)
      END
C
C
C
      SUBROUTINE OUTYPR
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      WRITE (6,99999)
      WRITE (6,99998)
      IPAR = 0
      DO 20 I=1,NPARAM
           A = DSQRT(PARCOV(I,I))
           YDUM(I) = A
           IF (IFPAR(I).NE.0) GO TO 10
           WRITE (6,99997) I, PARAM(I), A
           GO TO 20
   10      IPAR = IPAR + 1
           WRITE (6,99996) I, PARAM(I), A, IPAR
   20 CONTINUE
C
      IOFF = 0
      IALL = 0
      DO 40 I=1,NPARAM
           DO 30 J=1,I
                IF (PARCOV(I,I).NE.0.D0) IALL = IALL + 1
                IF (I.EQ.J) GO TO 40
                IF (PARCOV(J,I).NE.0.D0) IOFF = IOFF + 1
   30      CONTINUE
   40 CONTINUE
      IF (IOFF.NE.0) CALL OUTYV
      IF (IALL.NE.0) RETURN
      WRITE (6,99995)
      STOP
```

```
C
99999 FORMAT (40H0******COMPLETE LIST OF PARAMETER VALUES)
99998 FORMAT (36H0          PARAMETER      UNCERTAINTY, 6H     V,
     *     22HARIED PARAMETER NUMBER)
99997 FORMAT (I5, 2F14.6)
99996 FORMAT (I5, 2F14.6, I10)
99995 FORMAT (46H******ERROR.  COVARIANCE MATRIX FOR PARAMETERS,
     *     1H , 44HMUST BE INITIALIZED IN SUBROUTINE SETPAR.***,
     *     3H***)
      END
C
C
C
      SUBROUTINE OUTDAT
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DIMENSION IDDUM(1)
      EQUIVALENCE(IDDUM(1),SIG(1))
C
      WRITE (6,99999)
C
      II = 0
      DO 10 I=1,NDAT
           II = II + I
           DUM(I) = DSQRT(VARDAT(II))
   10 CONTINUE
C
      WRITE (6,99998)
      WRITE (6,99997) (I,E(I),DATA(I),DUM(I),I=1,NDAT)
C
      II = 0
      IOFF = 0
      DO 30 I=1,NDAT
           DO 20 J=1,I
                II = II + 1
                IF (I.EQ.J) GO TO 30
                IF (VARDAT(II).NE.0.D0) IOFF = IOFF + 1
   20      CONTINUE
   30 CONTINUE
      IF (IOFF.NE.0) CALL OUTV(VARDAT, DUM, IDDUM, NDAT)
      RETURN
C
99999 FORMAT (25H0****** INPUT DATA VALUES)
99998 FORMAT (46H0        DATA POINT        VALUE     UNCERTAIN,
     *     2HTY)
99997 FORMAT (I5, 3F14.6)
      END
```

```
C
C
C
      SUBROUTINE OUTTH
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      WRITE (6,99999)
      WRITE (6,99998)
      WRITE (6,99997) (I,E(I),DATA(I),TH(I),I=1,NDAT)
      RETURN
C
99999 FORMAT (30H0***** THEORETICAL CALCULATION)
99998 FORMAT (45H0           ENERGY         DATA            THEORY)
99997 FORMAT (I5, 3F14.6)
      END
C
C
C
      SUBROUTINE OUTG
C
C *** PURPOSE -- OUTPUT E VS G
C
C       NML, AUGUST 1980
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DATA T2 /10H   ENERGY  /
C
      WRITE (6,99999)
C
      MIN = 1
      MAX = MIN0(NPAR,7)
   10 WRITE (6,99998)
      WRITE (6,99997) T2, (I,I=MIN,MAX)
      DO 20 J=1,NDAT
           WRITE (6,99996) J, E(J), (G(J,I),I=MIN,MAX)
   20 CONTINUE
```

```
C
      IF (MAX.EQ.NPAR) GO TO 30
      MIN = MAX + 1
      MAX = MAX + 7
      IF (MAX.GT.NPAR) MAX = NPAR
      GO TO 10
   30 RETURN
C
99999 FORMAT (27H0****** PARTIAL DERIVATIVES)
99998 FORMAT (20X)
99997 FORMAT (10X, A10, I10, 6I14)
99996 FORMAT (I4, 2X, 1PG14.6, 2X, 7G14.5)
      END
C
C
C
      SUBROUTINE OUTP1
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
      WRITE (6,99999)
      WRITE (6,99998)
C
      DO 10 I=1,NPAR
           WRITE (6,99997) I, POLD(I), PARM(I)
   10 CONTINUE
      RETURN
C
99999 FORMAT (27H0***** INTERMEDIATE RESULTS)
99998 FORMAT (38H0        OLD PARAM       NEW PARAMETERS)
99997 FORMAT (I5, 5F14.6)
      END
C
C
C
      SUBROUTINE OUTP2
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
C
```

```
      WRITE (6,99999)
C
      II = 0
      DO 10 I=1,NPAR
           II = II + I
           PDUM(I) = DSQRT(VARNEW(II))
   10 CONTINUE
C
      WRITE (6,99998)
      WRITE (6,99997) (I,POLD(I),PARM(I),PDUM(I),I=1,NPAR)
C
      II = 0
      IOFF = 0
      DO 30 I=1,NPAR
           DO 20 J=1,I
                II = II + 1
                IF (I.EQ.J) GO TO 30
                IF (VARNEW(II).NE.0.D0) IOFF = IOFF + 1
   20      CONTINUE
   30 CONTINUE
      IF (IOFF.NE.0) CALL OUTV(VARNEW, PDUM, IDUM, NPAR)
      RETURN
C
99999 FORMAT (34H0******* NEW VALUES FOR PARAMETERS)
99998 FORMAT (46H0           POLD          PNEW       UNCERTAINTY)
99997 FORMAT (I5, 3F14.6)
      END
C
C
C
      SUBROUTINE OUTV(V, S, IC, N)
C
C *** PURPOSE -- OUTPUT TRIANGULAR VARIANCE V AS STD. DEV. AND CORRELATI
C
C       NML, AUGUST 1980
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION V(1), S(N), IC(N)
C
C
      WRITE (6,99999)
C
      NN = 0
      MANY = 25
      M = N/MANY
      MM = M*MANY
      IF (MM.NE.N) M = M + 1
      IMIN = 1 - MANY
      IMAX = MIN0(MANY,N)
```

```
      DO 50 J=1,M
           IMIN = IMIN + MANY
           MAX = (IMIN*(IMIN-1))/2
           WRITE (6,99998) (I,I=IMIN,IMAX)
           IMAX = IMAX + MANY
           IMAX = MIN0(IMAX,N)
           DO 40 I=IMIN,N
                MIN = MAX + IMIN
                MAX = MAX + I
                JMAX = I
                MM = JMAX - IMIN + 1
                IF (MM.GT.MANY) JMAX = IMIN + MANY - 1
                SI = S(I)
                IL = MIN - 1
                DO 10 L=IMIN,JMAX
                     IL = IL + 1
                     D = V(IL)*100.D0/(S(L)*SI)
                     IF (D.GT.0.D0) D = D + 0.5D0
                     IF (D.LT.0.D0) D = D - 0.5D0
                     IC(L) = D
   10           CONTINUE
                IF (J.EQ.1) GO TO 30
                NOT = 0
                JJ = JMAX
                IF (JMAX.EQ.I) JJ = JJ - 1
                JJP = IMIN
                IF (JJP.GT.JJ) GO TO 30
                DO 20 L=JJP,JJ
                     IF (IC(L).EQ.0) GO TO 20
                     NOT = 1
   20           CONTINUE
                IF (NOT.EQ.1) GO TO 30
                NN = 1
                GO TO 40
   30           WRITE (6,99997) I, SI, (IC(L),L=IMIN,JMAX)
   40      CONTINUE
   50 CONTINUE
      RETURN
C
99999 FORMAT (32H0       STD. DEV.    CORRELATION)
99998 FORMAT (1H0, 17X, 25I4)
99997 FORMAT (I4, 1PG14.6, 25I4)
      END
C
C
C
      SUBROUTINE OUTYV
C
C *** PURPOSE -- OUTPUT COVARIANCE PARCOV AS STD. DEV. AND CORRELATION
```

```
C
C       NML, JANUARY 1981
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /YRPAR/ PARAM(26), PARCOV(26,26), YDUM(26),
     *     IYDUM(26), IFPAR(26), IWHERE(26), NPARAM
C
      WRITE (6,99999)
C
      N = NPARAM
      NN = 0
      MANY = 25
      M = N/MANY
      MM = M*MANY
      IF (MM.NE.N) M = M + 1
      IMIN = 1 - MANY
      IMAX = MIN0(MANY,N)
      DO 50 J=1,M
           IMIN = IMIN + MANY
           WRITE (6,99998) (I,I=IMIN,IMAX)
           IMAX = IMAX + MANY
           IMAX = MIN0(IMAX,N)
           DO 40 I=IMIN,N
                JMAX = I
                MM = JMAX - IMIN + 1
                IF (MM.GT.MANY) JMAX = IMIN + MANY - 1
                SI = YDUM(I)
                DO 10 L=IMIN,JMAX
                     D = PARCOV(L,I)*100.D0/(YDUM(L)*SI)
                     IF (D.GT.0.D0) D = D + 0.5D0
                     IF (D.LT.0.D0) D = D - 0.5D0
                     IYDUM(L) = D
   10           CONTINUE
                IF (J.EQ.1) GO TO 30
                NOT = 0
                JJ = JMAX
                IF (JMAX.EQ.I) JJ = JJ - 1
                JJP = IMIN
                IF (JJP.GT.JJ) GO TO 30
                DO 20 L=JJP,JJ
                     IF (IYDUM(L).EQ.0) GO TO 20
                     NOT = 1
   20           CONTINUE
                IF (NOT.EQ.1) GO TO 30
                NN = 1
                GO TO 40
   30           WRITE (6,99997) I, SI, (IYDUM(L),L=IMIN,JMAX)
   40      CONTINUE
   50 CONTINUE
      RETURN
```

```
C
99999 FORMAT (32H0       STD. DEV.    CORRELATION)
99998 FORMAT (1H0, 17X, 25I4)
99997 FORMAT (I4, 1PG14.6, 25I4)
      END
C
C
C
      SUBROUTINE OUTREL
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      COMMON /PAR/ POLD(25), PARM(25), PDUM(25), VARPAR(325),
     *     VARNEW(325), IDUM(25)
      COMMON /DAT/ E(51), E2(51), DATA(51), VARDAT(1326),
     *     EN(1326), TH(51), DUM(51), SIG(51)
      COMMON /BOTH/ G(51,25), EMG(51,25)
      DIMENSION FORMT(4)
      DATA PL /1H(/, PR /1H)/
      DATA FORMT /40H((1X,4(A1,I3,A1,0PF8.0,1PG13.5,2X)))    /
      DATA AA5 /10H.1,1PG13.5/, AA4 /10H.2,1PG13.5/, AA3 /
     *     10H.3,1PG13.5/, AA2 /10H.4,1PG13.5/, AA1 /
     *     10H.5,1PG13.5/, AA0 /10H.6,1PG13.5/, AA6 /
     *     10H.0,1PG13.5/
      DATA T1 /10HENERGY    /, T2 /10HRESIDUAL  /
C
C
      WRITE (6,99999)
      GO TO 10
C
      ENTRY OUTREB
      WRITE (6,99998)
C
   10 WRITE (6,99997) (T1,T2,I=1,4)
      FORMT(3) = AA6
      IF (E(NDAT).LT.1.D5) FORMT(3) = AA5
      IF (E(NDAT).LT.1.D4) FORMT(3) = AA4
      IF (E(NDAT).LT.1.D3) FORMT(3) = AA3
      IF (E(NDAT).LT.1.D2) FORMT(3) = AA2
      IF (E(NDAT).LT.1.D1) FORMT(3) = AA1
      IF (E(NDAT).LT.1.D0) FORMT(3) = AA0
C
      N = NDAT
      M = N/4
      MM = M*4
      IF (MM.NE.N) GO TO 20
      MM = M
      GO TO 30
   20 M = M + 1
      K = MM + 4 - N
      IF (K.GE.M) GO TO 70
      MM = M - K
```

```
   30 CONTINUE
      IM = 3*M
      DO 40 I=1,MM
           IM = IM + 1
           WRITE (6,FORMT) (PL,L,PR,E(L),DUM(L),L=I,IM,M)
   40 CONTINUE
      IF (MM.EQ.M) GO TO 60
      IM = 2*M + MM
      MM = MM + 1
      DO 50 I=MM,M
           IM = IM + 1
           WRITE (6,FORMT) (PL,L,PR,E(L),DUM(L),L=I,IM,M)
   50 CONTINUE
   60 CONTINUE
      RETURN
   70 WRITE (6,FORMT) (PL,I,PR,E(I),DUM(I),I=1,N)
      RETURN
99999 FORMAT (39H0***** LEAST-SQUARES WEIGHTED RESIDUALS,
     *     31H AT FORMER VALUES OF PARAMETERS)
99998 FORMAT (34H0***** BAYESIAN WEIGHTED RESIDUALS, 8H AT FORM,
     *     23HER VALUES OF PARAMETERS)
99997 FORMAT (3H0  , 4(2X, A10, 2X, A10, 4X))
      END
C
C
C
      SUBROUTINE OUTCHL(CHI)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /NUMBER/ NDAT, NPAR, ITER, ITMAX, CONVER
      WRITE (6,99999) CHI
      N = NDAT - NPAR
      IF (N.LE.0) RETURN
      CHI = CHI/DFLOAT(N)
      WRITE (6,99998) CHI
      RETURN
C
      ENTRY OUTCHB(CHI)
      WRITE (6,99997) CHI
      N = NDAT - NPAR
      IF (N.LE.0) RETURN
      CHI = CHI/DFLOAT(N)
      WRITE (6,99998) CHI
      RETURN
C
99999 FORMAT (42H0***** LEAST-SQUARES CHI SQUARED AT FORMER,
     *     24H VALUES OF PARAMETERS IS, F15.6)
99998 FORMAT (45H0CHI SQUARED DIVIDED BY DEGREES OF FREEDOM IS,
     *     F15.6)
99997 FORMAT (37H0***** BAYESIAN CHI SQUARED AT FORMER,
     *     24H VALUES OF PARAMETERS IS, F15.6)
      END
```

```
C
C
C
      SUBROUTINE OUTGMG(EN, NDAT)
C
C *** PURPOSE -- PRINT ALL OF TRIANGULAR ARRAY EN EXPLICITLY,
C ***            FOR DEBUG PURPOSES ONLY
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION EN(1)
C
      MANY = 8
      M = NDAT/MANY
      MM = M*MANY
      IF (MM.NE.NDAT) M = M + 1
C
      IMIN = 1 - MANY
      DO 20 J=1,M
           IMIN = IMIN + MANY
           MAX = (IMIN*(IMIN-1))/2
C
           DO 10 I=IMIN,NDAT
                MIN = MAX + IMIN
                MAX = MAX + I
                MMAX = MAX
                MM = MAX - MIN + 1
                IF (MM.GT.MANY) MMAX = MIN + MANY - 1
                WRITE (28,99999) I, (EN(IL),IL=MIN,MMAX)
   10      CONTINUE
C
           WRITE (28,99998)
   20 CONTINUE
C
      RETURN
99999 FORMAT (I4, 8G14.6)
99998 FORMAT (1H0)
      END
```

INTERNAL DISTRIBUTION

1-2. L. S. Abbott
3. G. F. Auchampaugh (Consultant)
4. Z. W. Bell
5. J. L. Bledsoe
6. R. F. Carlton (Consultant)
7. H. P. Carter/G. E. Whitesides/ CS Library
8. J. G. Craven
9. J. W. Dabbs
10. J. K. Dickens
11. G. de Saussure
12. C. J. Emerson
13. C. Y. Fu
14. P. W. Gaffney
15. G. Goldstein (Consultant)
16. P. Greebler (Consultant)
17. D. M. Hetrick
18. J. T. Holdeman
19. D. J. Horen
20. C. H. Johnson
21. D. C. Larson
22-42. N. M. Larson
43. W. B. Lowenstein (Consultant)
44. I. Y. Lee
45. R. L. Macklin
46. F. C. Maienschein
47. B. F. Maskewitz
48. G. S. McNeilly
49. P. S. Meszaros
50. S. Y. Ohr
51. D. K. Olsen
52. R. W. Peelle
53. C. M. Perey
54. F. G. Perey
55. R. B. Perez
56. S. Raman
57. M. Rhoades-Brown
58. R. Satchler
59. R. O. Sayer
60. A. L. Sjoreen
61. R. E. Uhrig (Consultant)
62. L. W. Weston
63. R. Wilson (Consultant)
64. R. Winters (Consultant)
65. Central Research Library
66. K-25 Plant Library
67. ORNL Y-12 Technical Library Document Reference Section
68-69. Laboratory Records Department
70. ORNL Patent Office
71. Laboratory Records (RC)

EXTERNAL DISTRIBUTION

72. Pedro Andreo, Servicio de Fisica, Depto. de Radiologia, Hospital Clinico Universitario, Avda. Gomez Laguna, s/n., Zaragoza-9, ESPANA (Spain).
73. Allan D. Carlson, Neutron Measurements and Research, U.S. Department of Commerce, National Bureau of Standards, Washington, DC 20234.
74. Chief, Mathematics and Geoscience Branch, DOE, Washington, DC 20545.
75. DOE Oak Ridge Operations, Reactor Division, P. O. Box E, Oak Ridge, TN 37830.
76. DOE Division of Reactor Research and Development, Washington, DC 20545.
77. J. C. Garth, RADC/ESR, Hanscom Air Force Base, MA 01731.

78. J. A. Lorenc, Nuclear Design Engineer, Nuclear Studies and Safety Department, 700 University Avenue, Toronto, Ontario M5G 1X6.

79. D. W. Muir, MS-B243, Los Alamos National Laboratory, Los Alamos, NM 87545.

80. J. N. Rogers, Division 8324, Sandia Laboratories, Livermore, CA 94550.

81. Office of Assistant Manager for Energy Research and Development, Department of Energy, Oak Ridge Operations, Oak Ridge, TN 37830.

82-264. For distribution as shown in TID-4500 Distribution Category UC-79d, Liquid Metal Fast Breeder Reactor Physics - Base (60 copies - ENDF distribution).

☆U.S. GOVERNMENT PRINTING OFFICE: 1982-546-067/134