MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES
3 4456 0364504 5

ORNL-3259
UC-32 – Mathematics and Computers
TID-4500 (17th ed.)

A GENERALIZED LEAST SQUARES PROGRAM FOR THE IBM 7090 COMPUTER

M. H. Lietzke

CENTRAL RESEARCH LIBRARY
DOCUMENT COLLECTION
LIBRARY LOAN COPY
DO NOT TRANSFER TO ANOTHER PERSON
If you wish someone else to see this document, send in name with document and the library will arrange a loan.

OAK RIDGE NATIONAL LABORATORY
operated by
UNION CARBIDE CORPORATION
for the
U.S. ATOMIC ENERGY COMMISSION

ORNL-3259

Contract No. W-7405-eng-26

Chemistry Division

A GENERALIZED LEAST SQUARES PROGRAM FOR THE IBM 7090 COMPUTER

M. H. Lietzke

DATE ISSUED

MAR 21 1962

OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee
operated by
UNION CARBIDE CORPORATION
for the
U.S. ATOMIC ENERGY COMMISSION

MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES

3 4456 0364504 5

ABSTRACT

A generalized least squares program has been written for the IBM 7090 computer. The program is written in two parts: a subroutine to perform all matrix operations and furnish the calling program with the coefficients, standard errors in the coefficients, variance of fit, and a complete error analysis; and a calling program which reads input, calls the subroutine, and writes requested output. The subroutine may be used separately in any Fortran or FAP program where a least squares fit is needed. The user must furnish a subroutine to define the desired function and partial derivatives with respect to the coefficients.

A previous ORNL report describes POLFIT II, an IBM 7090 program for polynomial least squares fitting[1]. The present program will perform

(1) M. P. Lietzke, ORNL-3132 (April 24, 1961).

a least squares fit to any function, linear or non-linear, with up to eight coefficients and up to five independent variables. Weighting factors may either be defined or computed. The user must furnish a subroutine to define the function to be fitted and the partial derivatives with respect to the coefficients. The standard output of the calling program includes the values of the coefficients, the standard error in each coefficient, the variance of fit, and a point by point solution of the equation for each data point. The variance in the dependent variable is also computed for each data point. In addition, the inverse matrix may be printed out at the option of the user.

Since a mathematical discussion of least squares fitting was included in the POLFIT II report it will not be repeated here.

Use of Subroutine NLLS

The generalized least squares subroutine NLLS requires no common storage. It does require the library subroutines XLOC, MATEQ, and SQRT. These routines are standard library subroutines except for MATEQ which may be included with the binary deck. To use the subroutine with any FORTRAN program it is necessary to include the following statements:

DIMENSION W(250), X(250,5)*, Y(250), GP1(8), STDE(8), YCALC(250), VARY(250), A(8,8), B(8), XTX(8,8), TB(8), CONV(8), DERIV(8), C(8,8), VMAT(8,8)

CALL NLLS(W,X,Y,GP1,STDE, YCALC, VARY, NERR, EPS, NIT, NP, NDP, VARF, VARF1, C).

The following definitions apply:

W	Weighting factor (input, may be zero)
X	Independent variable*
Y	Dependent variable
GP1	Initial guesses on the coefficients
STDE	Standard error in each coefficient
YCALC	Calculated value of Y for each data point using the converged values of the coefficients
VARY	Variance in Y
NERR	Error indicator 1 = no error 2 = singular matrix 3 = non-convergence within the specified number of iterations 4 = singular inverse matrix
EPS	Maximum allowed percentage difference between successive values of the coefficients at convergence
NIT	Number of iterations to be allowed
NP	Number of coefficients ($1 \leq NP \leq 8$)
NDP	Number of data points
VARF	Variance of fit
VARF1	$VARF/\sum_i W_i$
C	Inverse matrix

* The independent variable is doubly subscripted since up to five independent variables may be specified.

The statement following the call of the subroutine NLLS should test the error indicator (NERR) and appropriate action should be taken if an error has occurred. This can conveniently be done with a computed GO TO. (See Calling Program in Appendix A.)

Note that the value of the independent variable is doubly subscripted. The first index refers to the number of the data point and the second to the specified independent variable. For example, X(J,1) would refer to the jth data point of the first independent variable and X(J,5) to the jth data point of the fifth independent variable.

Subroutine NLLS calls a subroutine SUBRT which must be supplied by the user. This subroutine must evaluate the function to be fitted, the partial derivatives of the dependent variable with respect to the coefficients, the difference between the observed and calculated value of the dependent variable, and a statement on the weight to be assigned to each data point.

Subroutine NLLS does not have built-in scaling. Scaling must be done by the user, if necessary, on the data.

Subroutine SUBRT

A sample subroutine to fit the function

$$y = a_o x_1 + a_1 x_2$$

is shown. Note that

$$\frac{\partial y}{\partial a_o} = x_1 \quad \text{and} \quad \frac{\partial y}{\partial a_1} = x_2$$

and assume that all points are assigned weights of unity.

```
C       TEST ROUTINE
        SUBROUTINE SUBRT(J,W,X,Y,GP1,DERIV,YC,F1,WI)
        DIMENSION W(250),X(250,5),Y(250),GP1(8),DERIV(8)
        YC=GP1(1)*X(J,1)+GP1(2)*X(J,2)
        F1=Y(J)-YC
        DERIV(1)=X(J,1)
        DERIV(2)=X(J,2)
        WI=1.0
        RETURN
        END
```

The following definitions apply:

J Index on data point

W Weighting factor (input, may be zero)

X Independent variable

Y Dependent variable (observed value)

GP1 Guess on value of coefficient. Thus GP1(1)=a_o and GP1(2)=a_1 in the example above.

DERIV Partial derivative of dependent variable with respect to coefficient.

YC Value of dependent variable computed using the current guesses on the coefficients.

F1 Difference between the observed and calculated value of the dependent variable.

WI Weight assigned to data point.

In case specific weights are read in for each data point then the statement

```
WI = W(J)
```

would be made in the subroutine. In the event weights are to be computed, then the appropriate statement would be included in SUBRT. If all weights are unity then WI = 1.0.

Any subroutine SUBRT written would have the same arguments and dimension statement as in the example.

If for reasons of memory space the user desires to change the maximum number of data points which the program can handle, he should change all the dimensions which are currently 250 to the desired value and recompile.

Description of Calling Program

A calling program for the generalized least squares subroutine has been written which is compatible with the IBM FORTRAN Monitor System. It uses logical tape 10 for input and logical tape 9 for output. No other tapes are used.

The program provides a complete package for least squares fitting. Input is read from tape 10 in the form shown in figure 1 and output is written on tape 9 as shown in figure 2. The standard output includes everything except the inverse matrix, the print-out of which is optional. The inverse matrix is always listed as a square array.

The program checks for errors, writes out an appropriate error message if one is detected and goes on to try the next case. Input data are described in Table 1 and a sample is given in Figure 1. Note that the integer quantities are packed to the right. Floating point quantities must have the decimal point written, may have power of 10 given.

TABLE 1. INPUT DATA

Card No.	Field	Variable	Type	
0	1-48	Title	BCD	Up to 48 alphameric characters which will appear on each page of output.
	49-52	NDP	INT	Number of data points $NP < NDP \leq 250$.
	53-54	NP	INT	Number of coefficients $1 \leq NP \leq 8$.
	55-56	NX	INT	Number of independent variables $1 \leq NX \leq 5$.
	57-60	NIT	INT	Number of iterations allowed on a given case.
	61-70	EPS	Fl. Pt.	Epsilon, the maximum allowed fractional difference between successive values of the coefficients at convergence.
	71-72	NINV	INT	= 0, do not list inverse matrix; = 1, list inverse matrix.
1	1-10	GP1(1)	Fl. Pt.	Guess on first coefficient.
	11-20	GP1(2)	Fl. Pt.	Guess on second coefficient.
	21-30	GP1(3)	Fl. Pt.	Guess on third coefficient.
	⋮	⋮	⋮	⋮
	61-70	GP1(7)	Fl. Pt.	Guess on seventh coefficient.
(1A	1-10	GP1(8)	Fl. Pt.	Guess on eighth coefficient (omitted if $NP < 8$)).
2⟶NDP+1	1-10	W	Fl. Pt.	Weighting factor (may be omitted if defined in SUBRT).
	11-20	Y	Fl. Pt.	Dependent variable.
	21-30	X_1	Fl. Pt.	1st independent variable.
	31-40	X_2	Fl. Pt.	2nd independent variable.
	41-50	X_3	Fl. Pt.	3rd independent variable.
	51-60	X_4	Fl. Pt.	4th independent variable.
	61-70	X_5	Fl. Pt.	5th independent variable.

650 DATA SHEET

REQUEST _____ JOB TITLE _____ WRITTEN BY _____ DATE _____

1 - 10	11 - 20	21 - 30	31 - 40	41 - 50	51 - 60	61 - 70	71 - 80
TEST ROUTINE	M. H. LIETZKE				10 22 10	.001	1 0
GPI_1	GPI_2						
1.0	1.5						1
w	y	x_1	x_2				
	1.8	1.0	.5				2
	2.9	2.0	.6				3
	4.1	3.0	.7				4
	5.2	4.0	.8				5
	6.4	5.0	.9				6
	7.4	6.0	1.0				7
	8.7	7.0	1.1				8
	9.9	8.0	1.2				9
	11.0	9.0	1.3				10
	12.1	10.0	1.4				11

Figure 1. Input for Test Case

Figure 2

TEST ROUTINE M.H.LIETZKE

	PARAMETER	STD ERROR
A_1	9.95909E-01	1.56274E-02
A_2	1.55000E 00	9.77009E-02

VARF 3.27273E-03 VARF1 3.27273E-04

Y(OBS)	Y(CALC)	VAR(Y)	X(1)	X(2)
1.80000E 00	1.77091E 00	1.13058E-03	1.00000E 00	5.00000E-01
2.90000E 00	2.92182E 00	8.13223E-04	2.00000E 00	6.00000E-01
4.10000E 00	4.07273E 00	5.75207E-04	3.00000E 00	7.00000E-01
5.20000E 00	5.22364E 00	4.16529E-04	4.00000E 00	8.00000E-01
6.40000E 00	6.37455E 00	3.37190E-04	5.00000E 00	9.00000E-01
7.40000E 00	7.52545E 00	3.37190E-04	6.00000E 00	1.00000E 00
8.70000E 00	8.67636E 00	4.16529E-04	7.00000E 00	1.10000E 00
9.90000E 00	9.82727E 00	5.75207E-04	8.00000E 00	1.20000E 00
1.10000E 01	1.09782E 01	8.13224E-04	9.00000E 00	1.30000E 00
1.21000E 01	1.21291E 01	1.13058E-03	1.00000E 01	1.40000E 00

7.462120E-02	-4.583333E-01
-4.583333E-01	2.916666E 00

APPENDIX

The Fortran lists of the Generalized Least Squares Calling Program and Subroutine are reported here. The Fortran and/or Binary decks are available from the author.

```
C     NONLINEAR LEAST SQUARES CODE      M.H.LIETZKE
      DIMENSION TITLE(8),W(250),X(250,5),Y(250),GP(8),GP1(8),
     1STDE(8),YCALC(250),VARY(250),C(8,8)
   20 READ INPUT TAPE 10,1,(TITLE(I),I#1,8),NDP,NP,NX,NIT,EPS,NINV
    1 FORMAT(8A6,I4,2I2,I4,E10.0,I2)
      READ INPUT TAPE 10,2,(GP(I),I#1,NP)
    2 FORMAT(7E10.0)                                                  23120125
      DO 18 I#1,NDP
   18 READ INPUT TAPE 10,2,W(I),Y(I),(X(I,J),J#1,NX)
      DO 21 I#1,NP                                                    23120205
   21 GP1(I)#GP(I)                                                    23120207
      CALL NLLS(W,X,Y,GP1,STDE,YCALC,VARY,
     1NERR,EPS,NIT,NP,NDP,VARF,VARF1,C)
      GO TO (30,100,101,102),NERR
   30 WRITE OUTPUT TAPE 9,7,(TITLE(I),I#1,8)                          14800819
    7 FORMAT(1H18A6/32H0     PARAMETER        STD ERROR)              23120821
      DO 130 I#1,NP                                                   23120823
  130 WRITE OUTPUT TAPE 9,8,I,GP1(I),STDE(I)                          23120901
    8 FORMAT(3H0 A,I1,1PE14.5,E15.5)                                  23120903
      GO TO (50,51,52,53,54),NX
   50 WRITE OUTPUT TAPE 9,9,VARF,VARF1
      GO TO 55
   51 WRITE OUTPUT TAPE 9,10,VARF,VARF1
      GO TO 55
   52 WRITE OUTPUT TAPE 9,11,VARF,VARF1
      GO TO 55
   53 WRITE OUTPUT TAPE 9,12,VARF,VARF1
      GO TO 55
   54 WRITE OUTPUT TAPE 9,13,VARF,VARF1
   55 DO 56 I#1,NDP
   56 WRITE OUTPUT TAPE 9,14,Y(I),YCALC(I),VARY(I),
     1(X(I,J),J#1,NX)
    9 FORMAT(7H0  VARF,1PE15.5,10H     VARF1,E15.5/
     152H0    Y(OBS)        Y(CALC)        VAR(Y)          X(1))
   10 FORMAT(7H0  VARF,1PE15.5,10H     VARF1,E15.5/
     152H0    Y(OBS)        Y(CALC)        VAR(Y)          X(1)
     214H          X(2))
   11 FORMAT(7H0  VARF,1PE15.5,10H     VARF1,E15.5/
     152H0    Y(OBS)        Y(CALC)        VAR(Y)          X(1)
     228H          X(2)          X(3))
   12 FORMAT(7H0  VARF,1PE15.5,10H     VARF1,E15.5/
     152H0    Y(OBS)        Y(CALC)        VAR(Y)          X(1)
     242H          X(2)          X(3)          X(4))
```

```
 13 FORMAT(7H0  VARF,1PE15.5,10H     VARF1,E15.5/
   152H0    Y(OBS)        Y(CALC)        VAR(Y)          X(1)
   256H          X(2)          X(3)          X(4)          X(5))
 14 FORMAT(1H01P8E14.5)
    IF(NINV)60,20,60
 60 WRITE OUTPUT TAPE 9,15
 15 FORMAT(//20H0     INVERSE MATRIX)
    DO 61 I#1,NP
 61 WRITEOUTPUTTAPE9,16,(C(I,J),J#1,NP)
 16 FORMAT(1H01P8E14.6)
    GO TO 20                                                          23121007
100 WRITE OUTPUT TAPE 9,19,(TITLE(I),I#1,8)
 19 FORMAT(1H18A6)
    WRITE OUTPUT TAPE 9,150
150 FORMAT(16H0SINGULAR MATRIX)
    GO TO 20
101 WRITE OUTPUT TAPE 9,19,(TITLE(I),I#1,8)
    WRITE OUTPUT TAPE 9,151,NIT
151 FORMAT(20H0DID NOT CONVERGE IN14,
   111H ITERATIONS)
    GO TO 20
102 WRITE OUTPUT TAPE 9,19,(TITLE(I),I#1,8)
    WRITE OUTPUT TAPE 9,152
152 FORMAT(24H0SINGULAR INVERSE MATRIX)
    GO TO 20
    END                                                               23121009
```

```
C     NONLINEAR LEAST SQUARES SUBROUTINE       M.H.LIETZKE                14800101
      SUBROUTINE NLLS(W,X,Y,GPI,STDE,YCALC,VARY,
     1NERR,EPS,NIT,NP,NDP,VARF,VARFI,C)
      DIMENSION W(250),X(250,5),Y(250),GPI(8),
     1STDE(8),YCALC(250),VARY(250),
     2A(8,8),B(8),XTX(8,8),TB(8),CONV(8),DERIV(8),C(8,8),                14800109
     3VMAT(8,8)                                                          14800110
      NERR#1                                                             14800604
      DO 118 N#1,NIT                                                     23120213
      DO 110 L#1,NP                                                      23120219
      B(L)#0.0                                                           23120221
      DO 110 M#1,NP                                                      23120223
  110 A(L,M)#0.0                                                         23120225
      DO 111 J#1,NDP                                                     23120301
      CALL SUBRT(J,W,X,Y,GPI,DERIV,YC,FI,WI)                             14800303
   50 DO 111 L#1,NP                                                      23120305
      B(L)#B(L)+DERIV(L)*FI*WI                                           23120307
      DO 111 M#L,NP                                                      23120309
  111 A(L,M)#A(L,M)+DERIV(L)*DERIV(M)*WI                                 23120311
      DO 112 M#2,NP                                                      23120313
      K#M-1                                                              23120315
      DO 112 I#1,K                                                       23120317
  112 A(M,I)#A(I,M)                                                      23120319
      DO 212 L#1,NP                                                      23120321
      DO 212 M#1,NP                                                      23120323
  212 XTX(L,M)#A(L,M)                                                    23120325
      LA#XLOCF(A(1,1))                                                   23120401
      LB#XLOCF(B(1))                                                     23120403
      DET#MATEQF(LA,LB,NP,1,8,8)                                         23120405
      IF(ABSF(DET)-1.0E-30)113,113,114                                   23120501
  113 NERR#2
      GO TO 200                                                          14800506
  114 DO 115 I#1,NP                                                      23120509
  115 TB(I)#GPI(I)+B(I)                                                  23120511
      DO 116 I#1,NP                                                      23120513
      CONV(I)#ABSF(GPI(I)/TB(I)-1.0)                                     23120515
      IF(CONV(I)-EPS)116,116,117                                         23120517
  116 CONTINUE                                                           23120519
      GO TO 120                                                          23120521
  117 DO 118 I#1,NP                                                      23120523
  118 GPI(I)#TB(I)                                                       23120525
      NERR#3
      GO TO 200                                                          14800604
```

```
120 DO 121 I#1,NP                                                23120605
121 GP1(I)#TB(I)                                                 23120607
    SUMW#0.0                                                     23120613
    VARF#0.0                                                     23120615
    DO 122 J#1,NDP                                               23120617
    CALLSUBRT(J,W,X,Y,GP1,DERIV,YC,F1,WI)                        14800618
    VARF#VARF+WI*F1**2
122 SUMW#SUMW+WI                                                 23120621
    VARF#VARF/(FLOATF(NDP)-FLOATF(NP))                           23120623
    VARF1#VARF/SUMW                                              23120625
    DO 125 I#1,NP                                                23120701
    DO 125 J#1,NP                                                23120703
    IF(I-J)124,123,124                                           23120705
123 C(I,J)#1.0                                                   23120707
    GOTO125                                                      23120709
124 C(I,J)#0.0                                                   23120711
125 CONTINUE                                                     23120713
    LA#XLOCF(XTX(1,1))                                           23120715
    LB#XLOCF(C(1,1))                                             23120717
    DET#MATEQF(LA,LB,NP,NP,8,8)                                  23120719
    IF(ABSF(DET)-1.0E-30)126,126,127                             23120721
126 NERR#4
    GO TO 200                                                    14800725
127 DO 128 I#1,NP                                                23120803
    DO 128 J#1,NP                                                23120805
128 VMAT(I,J)#C(I,J)*VARF                                        23120807
    DO 129 I#1,NP                                                23120809
129 STDE(I)#SQRTF(VMAT(I,I))                                     23120813
    DO 136 J#1,NDP                                               14800913
    CALL SUBRT(J,W,X,Y,GP1,DERIV,YC,F1,WI)                       14800915
    VY#0.0                                                       23120917
    YCALC(J)#YC                                                  14800920
    DO 135 I#1,NP                                                23120921
    DO 135 K#1,NP                                                23120923
135 VY#VY+DERIV(I)*DERIV(K)*VMAT(I,K)
136 VARY(J)#VY
200 RETURN                                                       14801003
    END                                                          14800007
```

ACKNOWLEDGEMENT

The author wishes to thank Marjorie P. Lietzke and Dr. R. W. Stoughton for helpful suggestions in the development of this program.

ORNL-3259
UC-32 — Mathematics and Computers
TID-4500 (17th ed.)

INTERNAL DISTRIBUTION

1. Biology Library
2. Reactor Division Library
3-4. Central Research Library
5. ORNL - Y-12 Technical Library, Document Reference Section
6-25. Laboratory Records Department
26. Laboratory Records, ORNL R.C.
27. L. G. Alexander
28. N. B. Alexander
29. C. L. Allen
30. P. R. Ammann
31. V. E. Anderson
32. D. E. Arnurius
33. G. J. Atta
34. S. E. Atta
35. L. L. Bennett
36. H. W. Bertini
37. N. A. Betz
38. J. P. Boatman
39. A. A. Brooks
40. G. L. Broyles
41. L. L. Bumgarner
42. T. H. Burke
43. W. R. Busing
44. E. B. Carter
45. H. P. Carter
46. A. Chetham-Strode
47. H. C. Claiborne
48. R. R. Coveyou
49. J. L. Cox
50. A. H. Culkowski
51. M. H. Davis
52. N. M. Dismuke
53. A. C. Downing
54. J. D. Drake
55. H. B. Eldridge
56. L. C. Emerson
57. M. B. Emmett
58. R. V. Fehling
59. M. Feliciano
60. G. Findley
61. B. A. Flores
62. T. B. Fowler
63. H. E. Francis
64. Bob Funderlic
65. J. L. Gabbard
66. D. A. Gardiner
67. W. Gautschi
68. A. A. Gose
69. A. A. Grau
70. C. D. Griffies
71. W. L. Griffith
72. E. E. Gross
73. C. E. Guthrie
74. J. Halperin
75. M. T. Harkrider
76. A. S. Householder
77. F. C. Hutton
78. John Innes
79. W. H. Jordan
80. P. R. Kasten
81. D. R. Kellogg
82. J. P. Kelly (K-25)
83. M. T. Kelley
84. E. M. Kidd
85. L. J. King
86. W. E. Kinney
87. G. B. Knight
88. C. E. Larson
89. J. G. LaTorre
90. M. E. LaVerne
91. D. A. Leiti
92. H. A. Levy
93-113. M. H. Lietzke
114. M. P. Lietzke
115. R. S. Livingston
116. E. C. Long
117. Jim Lucius
118. R. N. Lyon
119. M. J. Mader
120. D. S. Marion
121. B. F. Maskewitz (K-25)
122. M. C. McCord
123. K. H. McCorkle
124. J. W. McGraw
125. A. A. Miller

126. R. V. Miskell
127. H. S. Moran
128. C. W. Nestor
129. H. C. Owens
130. S. S. Parish
131. F. L. Parker
132. I. R. Parsley
133. S. K. Penny
134. A. M. Perry
135. W. D. Plengey
136. C. A. Preskitt
137. W. S. Pritchard
138. Mozelle Rankin
139. J. J. Rayburn
140. J. Replogle
141. William Riggle
142. R. H. Ritchie
143. M. T. Robinson
144. J. E. Rowe
145. R. M. Rush
146. Walter Russell
147. Ruth Sanders
148. E. G. Silver
149. F. J. Stanek
150. H. G. Stanley
151. R. W. Stoughton
152. J. G. Sullivan
153. H. E. Thompson
154. M. L. Tobias
155. D. K. Trubey
156. E. H. Walker
157. A. M. Weinberg
158. G. E. Whitesides
159. B. D. Williams
160. C. S. Williams
161. E. C. Witt
162. F. J. Witt
163. P. B. Wood
164. H. L. Yakel, Jr.
165. H. Zeldes
166. C. D. Zerby
167. W. Zobel
168. MIT Practice School
169-177. SHARE Library (K-25)

EXTERNAL DISTRIBUTION

178. Division of Research and Development, AEC, ORO
179-784. Given distribution as shown in TID-4500 (17th ed.) under Mathematics and Computers category (75 copies - OTS)