MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES
3 4456 0364379 7

ORNL-3132
UC-32 - Mathematics and Computers

POLFIT II, AN IBM 7090 PROGRAM FOR POLYNOMIAL LEAST SQUARES FITTING

M. P. 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

Printed in USA. Price $0.75. Available from the

Office of Technical Services
Department of Commerce
Washington 25, D.C.

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

ORNL-3132

Contract No. W-7405-eng-26

Mathematics Panel

POLFIT II, AN IBM 7090 PROGRAM FOR POLYNOMIAL LEAST SQUARES FITTING

M. P. Lietzke

DATE ISSUED

April 24, 1961

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 0364379 7

CONTENTS

ABSTRACT iii

INTRODUCTION 1

MATHEMATICAL DISCUSSION 2

USE OF SUBROUTINE 6

DESCRIPTION OF CALLING PROGRAM 7

APPENDIX 16

ACKNOWLEDGEMENT 21

ABSTRACT

A program has been written to perform polynomial least squares fits on the IBM 7090 computer. Weighting factors may be included with the data if desired. The program was written in two parts: a subroutine which does the fit and furnishes the calling program with the coefficients, standard errors in the coefficients, standard error of fit, and information necessary to compute 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.

INTRODUCTION

In analyzing experimental data it is often necessary to make a least squares fit and use the resulting coefficients in further computations. A polynomial least squares fitting subroutine was written to facilitate this procedure by eliminating both hand transcription of coefficients and the time lag caused by extra machine runs. The first version of the subroutine was written for an 8K IBM 704[1]. The present subroutine is an improved version written for the IBM 7090, which will permit use of weighting factors and will fit up to a seventh degree polynomial.

In order to make the Polfit II subroutine immediately available for least squares fitting a calling program has been written to read in data, call the subroutine to make the fit, and write the coefficients, standard errors in the coefficients and the standard error of fit on an output tape. In addition the user may request optional output which includes (1) a point by point solution of the equation for each input value of the independent variable, together with the difference between the observed and calculated values of the dependent variable and the variance at each point; (2) solutions of the equation and the variance for any number of values of the independent variable which may be supplied following the standard input; and (3) a listing of the inverse matrix $(X^TX)^{-1}$, where X is the observation matrix.

MATHEMATICAL DISCUSSION

To make a least squares fit to an equation $y = f(x, a_0, a_1, \ldots a_n)$ it is necessary to set up n "normal equations" for the fit. A brief discussion of the derivation of these equations may be found in Margenau and Murphy[2]. These n equations are then solved simultaneously to determine the coefficients $a_0, a_1, \ldots a_n$ such that

$$\sum_i (y_i(\text{observed}) - y_i(\text{calc}))^2 \text{ is a minimum.}$$

The equations may be set up and solved by the matrix methods described in the following discussion.

In the general case of a least squares fit to an equation $y = f(x, a_0, a_1, \ldots a_n)$ which is linear with respect to the coefficients $a_0, a_1, \ldots a_n$ an observation matrix X is set up with a row of partial derivatives evaluated for each data point i, i = 1, 2, ... K.

Thus:

$$X = \begin{pmatrix} \left(\frac{\partial y}{\partial a_0}\right)_{x_1} & \left(\frac{\partial y}{\partial a_1}\right)_{x_1} & \cdots & \left(\frac{\partial y}{\partial a_n}\right)_{x_1} \\ \left(\frac{\partial y}{\partial a_0}\right)_{x_2} & \left(\frac{\partial y}{\partial a_1}\right)_{x_2} & \cdots & \left(\frac{\partial y}{\partial a_n}\right)_{x_2} \\ \vdots & \vdots & & \vdots \\ \left(\frac{\partial y}{\partial a_0}\right)_{x_K} & \left(\frac{\partial y}{\partial a_1}\right)_{x_K} & \cdots & \left(\frac{\partial y}{\partial a_n}\right)_{x_K} \end{pmatrix}$$

This matrix is then transposed to give

$$X^T = \begin{pmatrix} \left(\frac{\partial y}{\partial a_0}\right)_{x_1} & \left(\frac{\partial y}{\partial a_0}\right)_{x_2} & \cdots & \left(\frac{\partial y}{\partial a_0}\right)_{x_K} \\ \left(\frac{\partial y}{\partial a_1}\right)_{x_1} & \left(\frac{\partial y}{\partial a_1}\right)_{x_2} & \cdots & \left(\frac{\partial y}{\partial a_1}\right)_{x_K} \\ \vdots & \vdots & & \vdots \\ \left(\frac{\partial y}{\partial a_n}\right)_{x_1} & \left(\frac{\partial y}{\partial a_n}\right)_{x_2} & \cdots & \left(\frac{\partial y}{\partial a_n}\right)_{x_K} \end{pmatrix}.$$

The observation matrix X is then augmented by the column vector of observed values of y to form X^A.

If weighting factors are to be used a matrix W is set up which has the weight associated with each point on the principal diagonal and zeros in the off diagonal positions, giving

$$W = \begin{pmatrix} w_1 & 0 & 0 & --- & 0 \\ 0 & w_2 & 0 & --- & \vdots \\ 0 & 0 & w_3 & & \vdots \\ \vdots & \vdots & & & \vdots \\ 0 & 0 & \cdots & \cdots & w_K \end{pmatrix}.$$

Now the matrix multiplication

$$X^T \, W \, X^A = A$$

will give the matrix A which is the matrix of the coefficients of a_0, a_1, ... a_n

in the "normal equations," and the vector B which is the vector of constant terms in the "normal equations."

$$A = \begin{pmatrix} \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_0}\right)^2_{x_i} & \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_0}\right)_{x_i}\left(\frac{\partial y}{\partial a_1}\right)_{x_i} & \cdots & \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_0}\right)_{x_i}\left(\frac{\partial y}{\partial a_n}\right)_{x_i} \\ \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_1}\right)_{x_i}\left(\frac{\partial y}{\partial a_0}\right)_{x_i} & \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_1}\right)^2_{x_i} & \cdots & \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_1}\right)_{x_i}\left(\frac{\partial y}{\partial a_n}\right)_{x_i} \\ \vdots & & & \\ \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_n}\right)_{x_i}\left(\frac{\partial y}{\partial a_0}\right)_{x_i} & \cdots & \cdots & \sum_{i=1}^{K} w_i \left(\frac{\partial y}{\partial a_n}\right)^2_{x_i} \end{pmatrix}$$

$$B = \begin{pmatrix} \sum_{i=1}^{K} w_i \, y_i \left(\frac{\partial y}{\partial a_0}\right)_{x_i} \\ \sum_{i=1}^{K} w_i \, y_i \left(\frac{\partial y}{\partial a_1}\right)_{x_i} \\ \vdots \\ \sum_{i=1}^{K} w_i \, y_i \left(\frac{\partial y}{\partial a_n}\right)_{x_i} \end{pmatrix},$$

The values of a_0, a_1, ... a_n may now be calculated by solving the matrix equation AX = B, where A and B are the matrix and vector given above and X represents the unknowns a_0, a_1, ... a_n.

In the special case of a polynomial equation the partial derivatives

are known to be as follows: $\frac{\partial y}{\partial a_0} = 1$; $\frac{\partial y}{\partial a_1} = x$; ... $\frac{\partial y}{\partial a_n} = x^n$. Therefore the observation matrix may be written directly as

$$X = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_K & x_K^2 & \cdots & x_K^n \end{pmatrix} .$$

From this it can be seen that the matrix A and vector B will be as follows:

$$A = \begin{pmatrix} \sum_i w_i & \sum_i w_i\, x_i & \cdots & \sum_i w_i\, x^n \\ \sum_i w_i\, x_i & \sum_i w_i\, x_i^2 & \cdots & \sum_i w_i\, x_i^{(n+1)} \\ \vdots & \vdots & & \vdots \\ \sum_i w_i\, x_i^n & \sum_i w_i\, x_i^{(n+1)} & \cdots & \sum_i w_i\, x_i^{2n} \end{pmatrix}$$

$$B = \begin{pmatrix} \sum_i w_i\, y_i \\ \sum_i w_i\, y_i\, x_i \\ \sum_i w_i\, y_i\, x_i^2 \\ \vdots \\ \sum_i w_i\, y_i\, x_i^n \end{pmatrix}$$

The polfit program takes advantage of this fact to reduce computing time by setting up A and B directly from the input instead of going through all the preceding matrix operations.

USE OF SUBROUTINE

The Polfit II subroutine requires no common storage and uses no input-output equipment. It does require the library subroutines EXP(2, XLOC, MATEQ, and SQRT. These routines are standard library subroutines except for MATEQ which is included with the binary deck. To use the subroutine with any FORTRAN program it is necessary to include the following statements:

```
DIMENSION A(8), SEC(8), XTXI(8,8), VARC(8,8), X(250), Y(250), W(250)
CALL POLFIT (X, Y, W, K, N, A, SEC, SEF, XTXI, VARC, NERR)
```

The following definitions apply:

X	Independent variable
Y	Dependent variable
W	Weighting factor (may be left = 0)
K	Number of data points
N	Degree of polynomial $1 \leq N \leq 7$
A	Coefficients $a_0 \ldots a_{N+1}$
SEC	Standard error in each coefficient
SEF	Standard error of fit: $\sqrt{\sum w_i \, r_i^2/(K-N+1)}$
XTXI	Inverse Matrix: $(X^T X)^{-1}$
VARC	Variance Matrix $(\sum w_i \, r_i^2/(K-N+1) \, (X^T X)^{-1})$

NERR	Error indicator:	-1 = Too many data points: K > 250
		0 = Calculation all right
		+1 = Singular matrix.

The statement following the call should test the error indicator (NERR) and appropriate action should be taken if an error has occurred.

It is not necessary to supply weighting factors as the subroutine will automatically supply a value of 1.0 when a zero weighting factor is encountered.

The values of X, Y, and W are left intact by the subroutine, except for the case of W = 0 which is changed to W = 1.0, so they are available for further computation.

DESCRIPTION OF CALLING PROGRAM

A calling program for POLFIT II 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 was written to provide a complete package for polynomial fitting. Input is read from tape 10 in the form shown in figures 1 and 2, and output is written on tape 9 as shown in figures 3 and 4. The minimum output from all cases consists of the coefficients, standard error in each coefficient, and standard error of fit. In addition the user may request, by use of integers I1, I2, and I3, (1) the calculated value of the dependent variable, y; the difference, $y_{obs} - y_{calc}$; and the variance in y for each value of the independent variable, x, used in the fit; (2) values of y and the variance in y for I2

additional values of x read in after the fit is made; and (3) the inverse matrix $(X^TX)^{-1}$.

The program checks for errors, and goes on to try the next case if an error is detected.

Input data is described in table 1 and samples are given in figures 1 and 2. Please note that integer quantities are packed to the right. Floating point quantities must have 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.
	49-54	K	INT	Number of data points $N < K \leq 250$.
	55-56	N	INT	Degree of polynomial $1 \leq N \leq 7$.
	57-58	I1	INT	= -1, min. output only; = 0, skip to I2; = +1, solve eq. for input x, then to I2.
	59-62	I2	INT	= 0, skip to I3; = + No.; read extra x_s and solve for y_s, then to I3.
	63-64	I3	INT	= 0 skip to next case; = 1, list $(X^TX)^{-1}$.
1-K	1-10	x	Fl. Pt.	Independent variable
	11-20	y	Fl. Pt.	Dependent variable
	21-30	W	Fl. Pt.	Weighting factor (may be omitted)
K+1 on	1-10	x	Fl. Pt.	I2 extra values of x for which
	11-20	x	Fl. Pt.	solutions are desired, 7 numbers
	⋮		Fl. Pt.	per card. Omitted if I2 = 0.
	61-70	x	Fl. Pt.	

650 DATA SHEET

REQUEST _____ JOB TITLE POLFIT WRITTEN BY M. P. LIETZKE DATE _____

1 - 10	11 - 20	21 - 30	31 - 40	41 - 50	51 - 60	61 - 70	71 - 80	
TEST CASE 1	- LINEAR	FIT -	M. LIETZKE		7 1 1	3 1		0
x	y	w						
1.0	.36							1
2.0	.46							2
3.0	.62							3
4.0	.71							4
5.0	.87							5
6.0	.97							6
7.0	1.13							7
0.5	10.5	12.0						8

WCX-2627 (9-56)

Figure 1. Input for Test Case Without Weighting Factors.

650 DATA SHEET

REQUEST _____ JOB TITLE POLFIT _____ WRITTEN BY M. P. LIETZKE _____ DATE _____

1 - 10	11 - 20	21 - 30	31 - 40	41 - 50	51 - 60	61 - 70	71 - 80	
TEST CASE	2 - LINEAR	FIT - M.LIETZKE			7 1 1	0 1		0
1.0	3.6	-1 2.0						1
2.0	4.6	-1 1.1						2
3.0	6.2	-1 0.9						3
4.0	7.1	-1 1.5						4
5.0	8.7	-1 2.2						5
6.0	9.7	-1 1.4						6
7.0	1.13	1.0						7

WCX-2627 (9-56)

Figure 2. Input for Test Case With Weighting Factors.

TEST CASE1- LINEAR FIT - M.LIETZKE ORDER OF FIT 1 NO. OF POINTS 7

COEFFICIENT	STANDARD ERROR
2.200000E-01	1.584358E-02
1.278571E-01	3.542731E-03

STANDARD ERROR OF FIT 1.874637E-02

Y CALC	VAR. Y	Y OBS - Y CALC	X
3.478572E-01	1.631623E-04	1.214284E-02	1.000000E 00
4.757143E-01	1.004076E-04	-1.571430E-02	2.000000E 00
6.035714E-01	6.275473E-05	1.642857E-02	3.000000E 00
7.314286E-01	5.020378E-05	-2.142857E-02	4.000000E 00
8.592857E-01	6.275473E-05	1.071430E-02	5.000000E 00
9.871428E-01	1.004076E-04	-1.714283E-02	6.000000E 00
1.115000E 00	1.631623E-04	1.500003E-02	7.000000E 00

Y	VAR. Y	X
2.839286E-01	2.039528E-04	5.000000E-01
1.562500E 00	5.804812E-04	1.050000E 01
1.754286E 00	8.534643E-04	1.200000E 01

INVERSE MATRIX

J / I	1	2
1	7.14286E-01	-1.42857E-01
2	-1.42857E-01	3.57143E-02

Figure 3. Output from Test Case 1.

TEST CASE2- LINEAR FIT - M.LIETZKE ORDER OF FIT 1 NO. OF POINTS 7

COEFFICIENT	STANDARD ERROR
2.235357E-01	1.501560E-02
1.270557E-01	3.435875E-03

STANDARD ERROR OF FIT 2.172500E-02

Y CALC	VAR. Y	Y OBS - Y CALC	X
3.505914E-01	1.454031E-04	9.408630E-03	1.000000E 00
4.776471E-01	8.894838E-05	-1.764709E-02	2.000000E 00
6.047028E-01	5.610410E-05	1.529719E-02	3.000000E 00
7.317585E-01	4.687030E-05	-2.175853E-02	4.000000E 00
8.588142E-01	6.124698E-05	1.118575E-02	5.000000E 00
9.858700E-01	9.923414E-05	-1.586997E-02	6.000000E 00
1.112926E 00	1.608318E-04	1.707432E-02	7.000000E 00

Figure 4. Output from Test Case 2.

TEST CASE2- LINEAR FIT - M.LIETZKE ORDER OF FIT 1 NO. OF POINTS 7

INVERSE MATRIX

J / I	1	2
1	4.77712E-01	-9.73254E-02
2	-9.73254E-02	2.50124E-02

Figure 4 (continued)

REFERENCES

(1) M. H. Lietzke and Marjorie P. Lietzke, Polfit, a 704 Program for Polynomial Least Squares Fitting, ORNL CF 60-3-129 (March 31, 1960).

(2) Henry Margenau and George M. Murphy, The Mathematics of Physics and Chemistry, D. Van Nostrand Company, Inc. (1956), p. 517.

APPENDIX

The Fortran lists of the Polfit II Calling Program and Subroutine are reproduced here. The Fortran and/or binary decks are available from the author.

```
C       CALLING PROGRAM FOR POLFIT II  (7090)
        DIMENSION A(8),SEC(8),XTXI(8,8),VARC(8,8),XI(7),IDENT(8),
       1TITLE(8),X(250),Y(250),W(250)
   50   READ INPUT TAPE 10,1,(TITLE(I),I#1,8),K,N,I1,I2,I3
        IF(K-250)51,51,1000
   51   IF(K)300,300,52
   52   DO 54 I#1,K
   54   READ INPUT TAPE 10,2,X(I),Y(I),W(I)
        WRITE OUTPUT TAPE 9,3,(TITLE(I),I#1,8),N,K
        LIM#N+1
        CALL POLFIT(X,Y,W,K,N,A,SEC,SEF,XTXI,VARC,NERR)
        IF(NERR)1000,62,55
   55   WRITE OUTPUT TAPE 9,4
        IF(I2)50,50,56
   56   NC#I2/7
   57   IF(NC*7-I2)58,59,59
   58   NC#NC+1
        GO TO 57
   59   DO 60 I#1,NC
   60   READ INPUT TAPE 10,2,XI(1)
        GO TO 50
 1000   WRITE OUTPUT TAPE 9,3,(TITLE(I),I#1,8),N,K
        WRITE OUTPUT TAPE 9,5
        DO 61 I#1,K
   61   READ INPUT TAPE 10,2,DUMMY
        IF(I2)50,50,56
   62   WRITE OUTPUT TAPE 9,6,(A(I),SEC(I),I#1,LIM)
        WRITE OUTPUT TAPE 9,7,SEF
        LINE#LIM+4
        IF(I1)50,63,66
   63   IF(I2)64,64,72
   64   IF(I3)50,50,83
   66   WRITE OUTPUT TAPE 9,8
        LINE#LINE+2
        DO 71 I#1,K
        YCALC#A(1)
        DO 67 J#2,LIM
   67   YCALC#YCALC+A(J)*X(I)**(J-1)
        DIFF#Y(I)-YCALC
        SUM1#VARC(1,1)
        SUM2#0.0
        DO 69 IV1#2,LIM
        L#(IV1-1)*2
        SUM1#SUM1+VARC(IV1,IV1)*X(I)**L
        SUM3#0.0
        DO 68 J#IV1,LIM
        L2#IV1+J-3
   68   SUM3#SUM3+VARC(IV1-1,J)*X(I)**L2
   69   SUM2#SUM2+SUM3
        VARY#SUM1+2.0*SUM2
        WRITE OUTPUT TAPE 9,9,YCALC,VARY,DIFF,X(I)
        LINE#LINE+1
        IF(LINE-22)71,71,70
   70   WRITE OUTPUT TAPE 9,3,(TITLE(II),II#1,8),N,K
        WRITE OUTPUT TAPE 9,8
        LINE#2
   71   CONTINUE
        GO TO 63
   72   WRITE OUTPUT TAPE 9,10
        LINE#LINE+2
   82   IF(I2-7)73,73,74
```

```
 73 NX#I2                                                              14800519
    SENSE LIGHT 1                                                      14800521
    GO TO 75                                                           14800523
 74 NX#7                                                               14800525
    I2#I2-7                                                            14800601
 75 READ INPUT TAPE 10,2,(XI(I),I#1,NX)                                14800603
    DO 81 IX#1,NX                                                      14800605
    YCALC#A(1)                                                         14800607
    DO 76 I#2,LIM                                                      14800609
 76 YCALC#YCALC+A(I)*XI(IX)**(I-1)                                     14800611
    SUM1#VARC(1,1)                                                     14800613
    SUM2#0.0                                                           14800615
    DO 79 IV1#2,LIM                                                    14800617
    L#(IV1-1)*2                                                        14800619
 77 SUM1#SUM1+VARC(IV1,IV1)*XI(IX)**L                                  14800621
    SUM3#0.0                                                           14800623
    DO 78 J#IV1,LIM                                                    14800625
    L2#IV1+J-3                                                         14800701
 78 SUM3#SUM3+VARC(IV1-1,J)*XI(IX)**L2                                 14800703
 79 SUM2#SUM2+SUM3                                                     14800705
    VARY#SUM1+2.0*SUM2                                                 14800707
    WRITE OUTPUT TAPE 9,9,YCALC,VARY,XI(IX)                            14800709
    LINE#LINE+1                                                        14800711
    IF(LINE-22)81,81,80                                                14800713
 80 WRITE OUTPUT TAPE 9,3,N,K                                          14800715
    WRITE OUTPUT TAPE 9,10                                             14800717
    LINE#2                                                             14800719
 81 CONTINUE                                                           14800721
    IF(SENSE LIGHT 1)64,82                                             14800723
 83 IF(LINE+LIM-20)85,85,86                                            14800725
 85 WRITE OUTPUT TAPE 9,3,(TITLE(I),I#1,8),N,K                         14800801
 86 WRITE OUTPUT TAPE 9,11                                             14800805
    IDENT(1)#1                                                         14800807
    DO 87 I#2,LIM                                                      14800809
 87 IDENT(I)#IDENT(I-1)+1                                              14800811
    WRITE OUTPUT TAPE 9,12,(IDENT(I),I#1,LIM)                          14800813
    WRITE OUTPUT TAPE 9,13                                             14800815
    DO 88 I#1,LIM                                                      14800817
 88 WRITE OUTPUT TAPE 9,14,IDENT(I),(XTXI(I,J),J#1,LIM)                14800819
    GO TO 50                                                           14800821
300 CALL EXIT                                                          14800823
  1 FORMAT(8A6,I6,2I2,I4,2I2)                                          14800901
  2 FORMAT(7E10.0)                                                     14800903
  3 FORMAT(1H18A6,17H     ORDER OF FIT I3,18H     NO. OF POINTS I7)    14800905
  4 FORMAT(16H0SINGULAR MATRIX)                                        14800907
  5 FORMAT(26H0MORE THAN 250 DATA POINTS)                              14800909
  6 FORMAT(33H0  COEFFICIENT     STANDARD ERROR/(1H0 1P2E16.6))        14800911
  7 FORMAT(22H0STANDARD ERROR OF FIT 1PE16.6)                          14800913
  8 FORMAT(1H0/58H0         Y CALC          VAR. Y    Y OBS - Y CALC   14800915
   1      X)                                                           14800917
  9 FORMAT(1H0 1P4E16.6)                                               14800919
 10 FORMAT(1H0/39H0          Y          VAR. Y          X)            14800921
 11 FORMAT(1H0/15H0INVERSE MATRIX)                                     14800923
 12 FORMAT(3H0 J I7,7I14)                                              14800925
 13 FORMAT(3H  I)                                                      14801001
 14 FORMAT(1H0I2,1P8E14.5)                                             14801003
    END(0,1,0,0,1)
```

```
C     POLYNOMIAL LEAST SQUARES FITTING SUBROUTINE  POLFIT II
      SUBROUTINE POLFIT(X,Y,W,K,N,A,SEC,SEF,XTXI,VARC,NERR)
      DIMENSION A(8),SEC(8),XTXI(8,8),VARC(8,8),SUM(24),XTX(8,8),
     1B(8,9),X(250),Y(250),W(250)
      IF(K-250)50,50,1000
 1000 NERR#-1
 1001 RETURN
   50 SCALEX#0.0
      SCALEY#0.0
      LIM#N+1
      DO 52 I#1,K
      SCALEX#MAX1F(SCALEX,ABSF(X(I)))
      SCALEY#MAX1F(SCALEY,ABSF(Y(I)))
      IF(W(I))51,51,52
   51 W(I)#1.0
   52 CONTINUE
      DO 53 I#1,24
   53 SUM(I)#0.0
      DO 54 I#1,K
      YC#Y(I)/SCALEY
      XC#X(I)/SCALEX
      SUM(24)#SUM(24)+YC**2*W(I)
      DO 54 J#1,LIM
      JA#J+J-1
      SUM(JA)#SUM(JA)+XC**(JA-1)*W(I)
      SUM(JA+1)#SUM(JA+1)+XC**(JA)*W(I)
   54 SUM(J+15)#SUM(J+15)+YC*XC**(J-1)*W(I)
      DO 58 I#1,LIM
      I2#I+I
      XTX(I,I)#SUM(I2-1)
      B(I,I+1)#1.0
      B(I,1)#SUM(I+15)
      IF(I-LIM)55,58,58
   55 J1#I+1
      DO 57 J#J1,LIM
      B(I,J+1)#0.0
      J2#J-J1+I2
      XTX(I,J)#SUM(J2)
      XTX(J,I)#SUM(J2)
   56 B(J,I+1)#0.0
   57 CONTINUE
   58 CONTINUE
      LA#XLOCF(XTX(1,1))
      LB#XLOCF(B(1,1))
      M#LIM+1
      DET#MATEQF(LA,LB,LIM,M,8,8)
      IF(DET-1.0E-35)59,59,60
   59 NERR#1
      GO TO 1001
   60 NERR#0
      DO 70 I#1,LIM
      A(I)#B(I,1)
      DO 70 J#1,LIM
   70 XTXI(I,J)#B(I,J+1)
      SIGR2#SUM(24)
      DO 61 I#1,LIM
   61 SIGR2#SIGR2-A(I)*SUM(I+15)
      VARF#SIGR2/FLOATF(K-LIM)
      SEF#SQRTF(VARF)*SCALEY
      DO 63 I#1,LIM
      DO 62 J#1,LIM
```

```
   62 VARC(I,J)#XTXI(I,J)*VARF                                          14800517
   63 SEC(I)#SQRTF(VARC(I,I))                                           14800519
      TEST#1.0                                                          14800521
      DO 64 I#1,LIM                                                     14800523
      A(I)#A(I)*SCALEY/TEST                                             14800525
      SEC(I)#SEC(I)*SCALEY/TEST                                         14800601
   64 TEST#TEST*SCALEX                                                  14800603
      DO 65 I#1,LIM                                                     14800605
      DO 65 J#1,LIM                                                     14800607
      TEST#SCALEX**(I+J-2)                                              14800609
      XTXI(I,J)#XTXI(I,J)/TEST                                          14800611
   65 VARC(I,J)#VARC(I,J)/TEST*SCALEY**2                                14800613
      GO TO 1001                                                        14800615
      END(0,1,0,0,1)
*DATA                                                                   14800009
```

ACKNOWLEDGEMENT

The author wishes to thank Drs. M. H. Lietzke and R. W. Stoughton for helpful suggestions in the development of this program.

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. H. H. Bottenbruch
40. A. A. Brooks
41. G. L. Broyles
42. L. L. Bumgarner
43. T. H. Burke
44. W. R. Busing
45. E. B. Carter
46. H. P. Carter
47. C. E. Center (K-25)
48. R. A. Charpie
49. A. Chetham-Strode
50. H. C. Claiborne
51. E. L. Cooper
52. R. R. Coveyou
53. J. L. Cox
54. A. H. Culkowski
55. M. H. Davis
56. N. M. Dismuke
57. A. C. Downing
58. J. D. Drake
59. H. B. Eldridge
60. L. C. Emerson
61. M. B. Emmett
62. R. V. Fehling
63. M. Feliciano
64. G. Findley
65. B. A. Flores
66. T. B. Fowler
67. H. E. Francis
68. Bob Funderlic
69. J. L. Gabbard
70. D. A. Gardiner
71. W. Gautschi
72. A. A. Gose
73. A. A. Grau
74. C. D. Griffies
75. W. L. Griffith
76. E. E. Gross
77. C. E. Guthrie
78. J. Halperin
79. M. T. Harkrider
80. A. S. Householder
81. F. C. Hutton
82. John Innes
83. W. H. Jordan
84. P. R. Kasten
85. D. R. Kellogg
86. J. P. Kelly (K-25)
87. M. T. Kelley
88. E. M. Kidd
89. L. J. King
90. W. E. Kinney
91. G. B. Knight
92. J. G. LaTorre
93. M. E. LaVerne
94. D. A. Leiti
95. H. A. Levy
96. M. H. Lietzke
97-116. M. P. Lietzke
117. R. S. Livingston
118. E. C. Long
119. Jim Lucius
120. R. N. Lyon
121. M. J. Mader
122. D. S. Marion
123. B. F. Maskewitz (K-25)
124. M. C. McCord
125. K. H. McCorkle
126. J. W. McGraw
127. A. A. Miller
128. R. V. Miskell

129. H. S. Moran
130. C. W. Nestor
131. H. C. Owens
132. S. S. Parish
133. F. L. Parker
134. I. R. Parsley
135. S. K. Penny
136. A. M. Perry
137. W. D. Plengey
138. C. A. Preskitt
139. W. S. Pritchard
140. Mozelle Rankin
141. J. J. Rayburn
142. J. Replogle
143. William Riggle
144. R. H. Ritchie
145. M. T. Robinson
146. J. E. Rowe
147. Walter Russell
148. Ruth Sanders
149. E. G. Silver
150. F. J. Stanek
151. H. G. Stanley
152. R. W. Stoughton
153. J. G. Sullivan
154. H. E. Thompson
155. M. L. Tobias
156. D. K. Trubey
157. E. H. Walker
158. D. J. Wehe
159. A. M. Weinberg
160. G. E. Whitesides
161. B. D. Williams
162. C. S. Williams
163. C. E. Winters
164. E. C. Witt
165. F. J. Witt
166. P. B. Wood
167. H. L. Yakel, Jr.
168. C. D. Zerby
169. W. Zobel
170. MIT Practice School
171-180. SHARE Library (K-25)

EXTERNAL DISTRIBUTION

181. Division of Research and Development, AEC, ORO
182. Martin Hochdorf, TVA, 116 Old Post Office, Chattanooga, Tennessee
183-797. Given distribution as shown in TID-4500 (16th ed.) under Mathematics and Computers category (75 copies – OTS)