AgRISTARS

Foreign Commodity Production Forecasting

FC-L0-00432 80-10275
JSC-16346
JUN 30 1980 NASA CR-
160728

A Joint Program for Agriculture and Resources Inventory Surveys Through Aerospace Remote Sensing

June 1980

STATISTICAL OUTLIER DETECTION (SOD): A COMPUTER PROGRAM FOR DETECTING OUTLIERS IN DATA

R. S. Chhikara, C. L. Horton, and M. A. Mendlowitz
Lockheed Engineering and Management Services Company, Inc.

and

A. H. Feiveson
National Aeronautics and Space Administration

(E80-10275) STATISTICAL OUTLIER DETECTION (SOD): A COMPUTER PROGRAM FOR DETECTING OUTLINERS IN DATA (Lockheed Engineering and Management) 25 p HC A02/MF A01 N80-30848

G3/43 Unclas 00275

LOCKHEED ENGINEERING AND MANAGEMENT SERVICES COMPANY, INC.
1830 NASA Road 1, Houston, Texas 77058

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
JSC-16346, FC-LO-00432		

4. Title and Subtitle
Statistical Outlier Detection (SOD): A Computer Program for Detecting Outliers in Data

5. Report Date
June 1980

6. Performing Organization Code

7. Author(s) R. S. Chhikara, C. L. Horton, M. A. Mendlowitz, Lockheed Engineering and Management Services Company, Inc. and A. H. Feiveson, National Aeronautics and Space Administration

8. Performing Organization Report No.
LEMSCO-14594

9. Performing Organization Name and Address
Lockheed Engineering and Management Services Company, Inc.
1830 NASA Road 1
Houston, Texas 77058

10. Work Unit No.

11. Contract or Grant No.
NAS 9-15800

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Houston, Texas 77058 Technical Monitor: R. O. Hill

13. Type of Report and Period Covered
Technical Report

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract
This document describes a computer program, SOD, for detecting outliers using a statistical test procedure. The user's guide, the program listing, and an example illustrating its use are included.

17. Key Words (Suggested by Author(s))
Outliers

18. Distribution Statement

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	23	

FC-LO-00432
JSC-16346

STATISTICAL OUTLIER DETECTION (SOD): A COMPUTER PROGRAM FOR DETECTING OUTLIERS IN DATA

Job Order 74-432

This report describes Sampling and Aggregation activities of the Foreign Commodity Production Forecasting project of the AgRISTARS program.

PREPARED BY

R. S. Chhikara, C. L. Horton, and M. A. Mendlowitz (Lockheed)
and
A. H. Feiveson (NASA)

APPROVED BY

NASA

R. O. Hill, Technical Monitor
Inventory Technology Section

LOCKHEED

B. L. Carroll, Manager
Commodity Forecasting Department

PREPARED BY

LOCKHEED ENGINEERING AND MANAGEMENT SERVICES COMPANY, INC.

Under Contract NAS 9-15800

For

Earth Observations Division

Space and Life Sciences Directorate

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

June 1980

LEMSCO-14594

CONTENTS

TABLES

1. DESCRIPTION

Described in this paper is a computer program that detects outliers in a univariate data set. This program, called SOD (statistical outliers detection), is capable of detecting as many as 19 outliers in a data set. It is written in FORTRAN and can be run in either an interactive or batch mode.

The SOD software consists of a main program and a subroutine. The main program (1) reads the data file, (2) writes the initial section of the report, and (3) iterates sequentially for testing the next number of potential outliers. The subroutine (1) calculates the test-critical values based on the number of potential outliers being tested and compares these to the observed values of test statistics and (2) gives the number of observations tested, total number of observations considered, mean, standard deviation, extreme observation, critical value, and computed test statistics. Also, it prints the number of observations declared as outliers and their values.

The number of potential outliers may be specified by the user or selected by the program. Though there is no limitation on the number of observations, it is not advisable to use it when there are more than 100 observations.

It is assumed that the set of observations is from a population which has a normal distribution. A significance level of 5 percent is assumed in developing the statistical test. If one or more observations do not conform to the hypothesis that all observations are from a common population, these observations are declared as outliers by the statistical test procedure. Refer to appendix A for a brief outline of the test procedure. Details of the procedure are described in references 4 and 6. The iterative procedure used for specifying the number of potential outliers is discussed in appendix A, and a program listing is given in appendix B.

It should be mentioned that the observations declared as outliers are not necessarily "bad" data points, but may be indicative of a nonnormal or multimodal distribution; hence, outlying observations should not necessarily be rejected, but must be treated cautiously.

2. USER GUIDE

The SOD program is operative on the PDP Support Processor at NASA/JSC, Houston.

To use the program, the user signs on to the computer by simultaneously depressing the control key (CNTR) and C key to begin the following dialogue:

MCR HEL [130,1] (carriage return)
TASK NAME>SOIL M (carriage return)
YOUR NAME>User's name (carriage return)[1]
MCR>PIP SOD.IN=DATA.DAT[2] (carriage return)
MCR>RUN SOD (escape)

The report will be written on the line printer.

---

[1] It may be necessary to simultaneously depress the CNTR and C keys.

[2] The data should have been keypunched previously and entered into a file (DATA.DAT) with the following format:

TITLE FOR REPORT
VALUE 1
VALUE 2
VALUE 3
VALUE 4
Each number should include a decimal point.
VALUE N

3. EXAMPLE

The following example illustrates the SOD program.

```
MCR>HEL [130.1]

TASK NAME>SOIL M

YOUR NAME>HORTON
MCR>PIP SOD.IN=DANIEL.DAT
MCR>RUN SOD$
 DANIEL (1959)

0.0
0.028
-.056
-.084
-.098
.126
.168
.196
.225
-.253
.295
-.309
.393
.407
.421
.435
.463
-.477
.547
.660
.744
-.744
-.758
-.814
-.814
-.898
1.080
-1.305
2.147
-2.666
-3.143

ENTER NUMBER OF ITERATIONS AS A 2 DIGIT NUMBER OR
ENTER BLANKS FOR DEFAULT = SQUARE OF NUMBER OF POINTS

SOD  -- STOP
```

TABLE 1.- COMPUTER OUTPUT FROM SOD

DANIEL (1959)

OBSERVATIONS:

0.0000	0.0280	-0.0560	-0.0840	-0.0980	0.1200
0.1680	0.1960	0.2250	-0.2530	0.2950	-0.3090
0.3930	0.4070	0.4210	0.4350	0.4630	-0.4770
0.5470	0.6600	0.7440	-0.7440	-0.7580	-0.8140
-0.8140	-0.8980	1.0800	-1.3050	2.1470	-2.6660
-3.1430					

NOPTS = 31

MEANS = -0.1317

STANDARD DEVIATION = 1.0001

NUMBER OF POTENTIAL OUTLIERS	TOTAL NUMBER OBSERVATIONS	MEAN	STANDARD DEVIATION	EXTREME OBSERVATION	ESD	CRITICAL TEST VALUES
6	31	-0.1317	1.0001	-3.1430	3.0111	3.3025
6	30	-0.0314	0.8435	-2.6660	3.1234	2.8205
6	29	0.0595	0.6932	2.1470	3.0116	2.6562
6	28	-0.0151	0.5754	-1.3050	2.2417	2.5568
6	27	0.0327	0.5266	1.0800	1.9842	2.4936
6	26	-0.0076	0.4930	-0.8980	1.8003	2.4406
5	31	-0.1317	1.0001	-3.1430	3.0111	3.2608
5	30	-0.0314	0.8435	-2.6660	3.1234	2.8049
5	29	0.0595	0.6932	2.1470	3.0116	2.5477
5	28	-0.0151	0.5754	-1.3050	2.2417	2.5509
5	27	0.0327	0.5266	1.0800	1.9842	2.4704
4	31	-0.1317	1.0001	-3.1430	3.0111	3.2076
4	30	-0.0314	0.8435	-2.6660	3.1234	2.7824
4	29	0.0595	0.6932	2.1470	3.0116	2.6340
4	28	-0.0151	0.5754	-1.3050	2.2417	2.5061
3	31	-0.1317	1.0001	-3.1430	3.0111	3.1362
3	30	-0.0314	0.8435	-2.6660	3.1234	2.7537
3	29	0.0595	0.6932	2.1470	3.0116	2.5539

OUTLIERS = 3

POINTS = -3.143 -2.666 2.147

4. REFERENCES

1. Barnett, V. D.; and Lewis, T.: Outliers in Statistical Data. Wiley, Chichester, (England), 1978.

2. Barnett, V. D.: The Study of Outliers: Purpose and Model. J. Appl. Statistics, vol. 17, no. 3, 1978, pp. 242-250.

3. Chhikara, R. S.: A Screening Procedure for Improving Large Area Crop Acreage Estimates. American Stat. Assn., 1979 proceedings of the section on Survey Research Methods, 1979, pp. 301-304.

4. Chhikara, R. S.; and Feiveson, A. H.: Extended Critical Values of Extreme Studentized Deviate Test Statistics for Detecting Multiple Outliers. Communications in Statistics, vol. B9, no. 2, 1980.

5. Daniel, C.: Use of Half-Normal Plots in Interpreting Factorial Two-Level Experiments. Technometrics, vol. 1, 1959, pp. 311-341.

6. Rosner, B.: On the Detection of Many Outliers. Technometrics, vol. 17, 1975, pp. 221-227.

APPENDIX A

AN ITERATIVE PROCEDURE FOR DETECTING MULTIPLE OUTLIERS

APPENDIX A

AN ITERATIVE PROCEDURE FOR DETECTING MULTIPLE OUTLIERS

A.1 THE STATISTICAL TEST

Given a set of N observations, suppose one wishes to test the null hypothesis of no outliers present in the data set against the alternative hypothesis that from one to k outliers exist, where k is specified in advance. This may be done by constructing a sequence of subsets of the data, $\{A_1, A_2, \cdots, A_k\}$, where A_1 is the full set of data and the subset A_{i+1} is formed by deleting from A_i the observation farthest away from the mean of A_i $(i = 1, 2, \cdots, k - 1)$. For each subset, the extreme studentized deviate (ESD) statistic is defined to be the maximum of the absolute values of the studentized residuals. (A studentized residual is the deviation of an observation from the sample mean divided by the sample standard deviation.)

Let t_i be the ESD from the i^{th} subset, A_i. Then corresponding to each t_i is a critical value λ_i such that either (a) $t_i < \lambda_i$ for $1 \leq i \leq k$, or (b) $t_i \geq \lambda_i$ for $1 \leq i \leq h$ and $t_i < \lambda_i$ for $i > h$ where $1 \leq h < k$. If (a) occurs, the hypothesis is accepted that there are no outliers; in case (b), the data are declared to have h outliers, with the observations deleted to form A_{h+1} as the outliers.

A.2 CHOICE FOR k

The critical values λ_i in SOD were constructed for the 5-percent significance level by numerical simulation using normally distributed data. These values are not independent of k; in fact, for a fixed value of i, they increase monotonically with k. As a consequence, even if the basic data are normally distributed, the power of the test against a fixed number of outliers decreases as k increases so that grossly overspecifying k may result in the failure to detect some or even all outliers. On the other hand, if k is underspecified, it is more likely that up to k outliers will be detected, but additional ones will not be found since the test assumes no more than k outliers are present. In either case, power is lost by using an inappropriate value of k.

If the underlying distribution of the data is nonnormal, especially if it is multimodal, the test will have a tendency to find many "outliers"; hence, if this situation occurs, even for fairly large k, one should be suspicious about the distribution of the data and look for underlying mechanisms which might have made the data multimodal or highly skewed.

The following two examples illustrate some of the above points.

Example 1: Soil moisture was measured at a depth interval of 5 to 9 centimeters for a wheat field near Colby, Kansas, on July 18, 1978. The following gravemetric measurements of water content in percentage of dry weight were obtained from 17 points within the field:

> 5.9, 6.4, 5.6, 7.5, 6.7, 4.0, 5.3, 5.5, 5.5, 3.5, 4.6, 10.5, 5.7, 7.3, 5.2, 9.7, 4.0

Two observations, 10.5 and 9.7, are suspicious, and one wants to know whether or not they could be regarded as outliers. An application of the test procedure with k = 2 declares these two observations as outliers; however, looking at the data and then choosing k alters the significance level of the test by an unpredictable amount. It is interesting to see what would have happened had k been chosen (in advance) as 1, 2, 3, or 4.

Table A-1 provides the computed ESD test statistic, t_i, and the corresponding 5-percent critical value, $\lambda_{i,k}$ ($i = 1, 2, \cdots, k$) for different cases.

TABLE A-1.- ESD STATISTICS AND CRITICAL VALUES

Subset	Extreme observation	ESD statistic (t_i)	Critical value ($\lambda_{i,k}$)			
			k = 1	k = 2	k = 3	k = 4
A_1	10.5	2.365	2.61	2.74	2.86	2.93
A_2	9.7	2.549		2.39	2.53	2.57
A_3	7.5	1.722			2.35	2.44
A_4	7.3	1.728				2.34
Number of outliers declared			0	2	2	0

From this table, it can be seen that, if the number of potential outliers were specified as either 1 or 4, none of the observations would have been declared as an outlier since $t_i < \lambda_{i,k}$ for all i. In the other two cases, $t_2 = 2.549$ exceeds both $\lambda_{2,2} = 2.39$ and $\lambda_{2,3} = 2.53$; hence, the two observations, 9.7 and 10.5, would be flagged as outliers. Thus, suspected observations may not be flagged as outliers by the test when k is under- or over-specified.

Example 2: The following soil moisture observations were obtained for a corn field near Colby, Kansas, on July 18, 1978, from the top soil layer (0- to 1-centimeter interval). In this case, gravemetric measurements of water content in percentages of dry weight were taken from 35 randomly selected points within the field:

11.5, 3.2, 19.2, 21.6, 5.7, 24.6, 2.1, 3.4, 4.4, 3.7, 4.2, 7.9,
7.1, 2.6, 3.5, 8.9, 1.8, 2.4, 6.0, 2.8, 29.2, 29.1, 19.6, 1.4,
4.4, 4.4, 2.9, 4.7, 3.2, 3.8, 2.6, 4.4, 4.6, 4.7, 4.6

At the top layer, soil moisture can be affected by a number of heterogeneous factors; thus, the observations which appear to be outliers may very well be legitimate.

Numerical ordering results in the following data set:

1.4, 1.8, 2.1, 2.4, 2.6, 2.6, 2.8, 2.9, 3.2, 3.2, 3.4, 3.5, 3.7, 3.8, 4.2, 4.4, 4.4, 4.4, 4.4, 4.6, 4.6, 4.7, 4.7, 5.7, 6.0, 7.1, 7.9, 8.9, 11.5, 19.2, 19.6, 21.6, 24.6, 29.1, 29.2

A quick glance shows that at least one significant gap occurs — between 11.5 and 19.2. Six observations are greater than or equal to 19.2. When the test procedure is applied assuming k = 6, these six observations are declared outliers. Furthermore, its repeated application with k = 2, 3, •••, 10 resulted in every additional extreme observation being flagged as an outlier. The flagged observations are 7.1, 7.9, 8.9, 11.5, 19.2, 19.6, 21.6, 24.6, 29.1, and 29.2. The first four of these observations should not be regarded as outliers since water content in this range was found to be quite reasonable for places at higher ground on the particular day of measurement. In this case, the blind applications of the test leads to the identification of false outliers. It must be recognized that these four observations simply cannot be lumped together with the remaining 25 observations and analyzed using data analysis techniques based on normal and/or unimodal models.

Ideally, one should not look at the data before specifying k. Rosner (1975), among others, suggested the use of a certain percentage of the number of observations to specify k. Barnett and Lewis (1978) proposed that a fractional power of N may be used for k. In the author's own work [Chhikara (1979)], the rule of $k = \sqrt{N}$, to the nearest integer, was employed and often proved quite satisfactory. Presently, this rule is extended to safeguard against errors of undetected outliers as described in the next section. While this modified method undoubtedly alters the signifiance level of the test, it still provides a useful device for screening data with no prior information. If outliers are detected using this procedure, one should not blindly accept such a declaration; instead, this should be taken as a starting point for further investigations about the cause of the suspected observations.

A.3 AN ITERATIVE METHOD

Start with $k = \sqrt{N}$ and compute the ESD test statistics. When the statistics t_i and the critical values $\lambda_{i,k}$ are compared, one of the following cases arises.

(a) $t_h \geq \lambda_{h,k}$, $h < k$, and $t_i < \lambda_{i,k}$, $(h + 1) \leq i \leq k$

(b) $t_i < \lambda_{i,k}$, $1 \leq i \leq k$

(c) $t_k \geq \lambda_{k,k}$

The test procedure described in section A.1 would declare h outliers in case (a), none in case (b), and k outliers in case (c). It is reasonable to assume that all potential outliers in the data were detected in case (a); and, hence, no further application of the procedure is needed. On the other hand, as we have seen, some outliers may have remained undetected in the other two cases; i.e., in case (b), the critical values $\lambda_{i,k}$ would have been smaller had a lesser number been specified for k; whereas, in case (c), additional extreme observations might have been declared as outliers had the test been made for more than k extreme observations. This suggests that an iterative procedure should be used to decrease k in case (b) and increase k in case (c). When k is decreased successively by one, a set of smaller critical values is being used, thus increasing the power of the test for declaring outliers. In case (b), the decision to stop iterating is made when $t_h \geq \lambda_{h,k}$ $(0 \leq h < k)$ which results in h outliers being declared. in case (c), In case (c), k is increased sucessively so that more extreme observations are tested as potential outliers. The test procedure is iterated until

$$t_{k+j} \geq \lambda_{k+j,k+j+1}$$

and

$$t_{k+j+1} < \lambda_{k+j+1,k+j+1}$$

with the number of iterations not to exceed a certain preset limit. The number of outliers declared is then set equal to $(k + j)$.

The critical values estimated by the SOD computer program are based on smoothing functions which fit the Monte Carlo-estimated critical values for $k = 1, 2, \cdots, 19$, and $N \leq 100$. Thus, one can apply the test procedure to detect as many as 19 outliers in a data set of up to 100 observations. However, as pointed out by Chhikara and Feiveson (1980), one would rarely need to consider testing for more than a few outliers. Since, at most, 50 percent of a set of observations can be thought of as outliers, the SOD program specifies $k < \min(I,19)$, where I is the largest integer less than or equal to N/2, for the upper bound on the number of iterations.

When using the iterative procedure to obtain k, the significance level of the test increases from 5 percent. Any such increase would depend upon the data size; however, it is insignificant for large samples since the critical values become insensitive to k in such situations.

A.4 EXAMPLES

Returning to Example 1 in section A.2, it is easily seen that the iterative test procedure will start by specifying $k = 4$ under the $\sqrt{N}$ rule and then will consider $k = 3$, since no outliers are declared in the first case. Because $t_2 > \lambda_{2,3}$, it will stop and declare two outliers for the data in Example 1.

In the case of Example 2, the procedure will continue iterating from the initial case of $k = 6$ to the last case of $k = 11$, declaring 10 outliers, since $t_{10} > \lambda_{10,11}$ and $t_{11} < \lambda_{11,11}$. However, as mentioned earlier, the 10 outlying observations are not necessarily outliers and flagging them using the test procedure reflects primarily on the data distribution being probably nonnormal and at least biomodal.

Example 3: Daniel (1959) reported the following data consisting of 31 contrasts in order of absolute value in a 2^5 experiment:

0.000, 0.028, -0.056, -0.084, -0.098, 0.126, 0.168, 0.196, 0.225, -0.253, 0.295, -0.309, 0.393, 0.407, 0.421, 0.435, 0.463, -0.477, 0.547, 0.660, 0.744, -0.744, -0.758, -0.814, -0.814, -0.898, 1.080, -1.305, 2.147, -2.666, and -3.143.

The test procedure started with k = 6 by the $\sqrt{N}$ rule and declared the last three points as outliers. The results, as output from the SOD computer program, are presented in table A-2 and show for each iteration: (a) the number of potential outliers specified; (b) the number of observations and the mean and standard deviation for each subset, (c) the extreme observation and the corresponding computed ESD statistic; (d) the 5-percent critical values; (e) the number of outliers declared; and (f) the outliers.

TABLE A-2.- STATISTICAL OUTLIER DETECTION (EXAMPLE 3)

Number of potential outliers	Number of observations	Mean	Standard deviation	Extreme observation	ESD	Critical value
6	31	-0.1317	1.0001	-3.1430	3.0111	3.3025
	30	-0.0314	0.8435	-2.6660	3.1234	2.8205
	29	0.0595	0.6932	2.1470	3.0116	2.6562
	28	-0.0151	0.5754	-1.3050	2.2417	2.5568
	27	0.0327	0.5268	1.0800	1.9882	2.4936
	26	-0.0076	0.4930	-0.8980	1.8063	2.4406

Number of outliers = 3

Outlying data points = -3.143, -2.666, 2.147

Interestingly, all three points, 2.147, -2.666, and -3.143 [which previously have been considered highly discordant on the null normal distribution; e.g., refer to Barnett (1978)], were flagged here as outliers even though the testing of a greater number of outliers was considered. Furthermore, the iterative procedure would have considered smaller values of k, thus increasing the power of the test, had these data points not been detected in the first instance.

A.5 CONCLUSION

The proposed iterative method deals with the problem of specifying the number of outliers being tested and should minimize the error that otherwise would occur in not detecting outliers when they exist in a data set. Although the chances of declaring false outliers may increase, the program still provides useful information for data screening prior to subsequent analysis.

Although the iterative test procedure is considered using the ESD test statistic, the basic approach can be adapted for any of the test statistics proposed in the literature; e.g., studentized range (STR), kurtosis (KUR), and R-statistic (RST). Details on these test statistics can be found in Rosner (1975).

It is desirable to determine the actual significance level reached by the iterative test procedure, particularly for small sample sizes. Also needed is an evaluation of its power against the non-null hypothesis of a smaller number of outliers. A theoretical solution is probably intractable; therefore, one should attempt to make these evaluations by using the Monte Carlo technique.

APPENDIX B

PROGRAM LISTING

APPENDIX B

PROGRAM LISTING

FORTRAN IV-PLUS V02-51
SOD.FTN /TR:BLOCKS/WR

```
      LOGICAL*1 INFO
      DIMENSION THETA(10),TITLE(15)
      COMMON ARG,T,AKK,AK1,BKK,BK1,NP,M(1000),D(1000),THEOK,THEOK1,
     1 EXPPK,EXPPK1,ISTEGT,OUTLIE(100)
      DIMENSION INFO(1000)
      DATA THETA /  5.425 , 1.213 , .3043 , 1.5302 , .6939 ,
     1             .45268 ,  .85333 , .491 , .2261 , .1429 /
C
C    ********  ASSIGN UNITS  *******
      CALL ASSIGN(1,'SOD.IN')
      CALL ASSIGN(2,'LP:')
      CALL ASSIGN(3,'TT:')
C
C       *********  INITIALIZE CONSTANTS  *******
 1050 CONTINUE
      S1=0.0
      S2=0.0
      DO 1080 I=1,1000
      D(I)=0.0
      M(I)=0
 1080 CONTINUE
 1090 CONTINUE
C
C  ************  READ DATA FROM INPUT FILE  **********
C  READ TITLE
      READ(1,1150) (TITLE(I),I=1,15)
 1150 FORMAT(15A4)
C
C   READ DATA POINTS
      I=0
 1200 CONTINUE
      I=I+1
      READ(1,1250,END=1500)D(I)
 1250 FORMAT(1F15.5)
      S1=S1+D(I)
      S2=S2+D(I)*D(I)
      GO TO 1200
C
C
C  *  CALCULATE AND WRITE NO. PTS, MEAN, STANDARD DEVIATION  **********
 1500 CONTINUE
      NOPTS=I-1
      PN=FLOAT(NOPTS)
      AVE=S1/PN
      VAR=(S2 - S1*S1/PN) / (PN-1.)
      SDEV=SQRT(VAR)
      WRITE(2,1550)(TITLE(I),I=1,15),(D(I),I=1,NOPTS)
 1550 FORMAT (//,10X,15A4,//,' OBSERVATIONS:',(/,6F11.4))
      WRITE(3,1580) (TITLE(I),I=1,15)
 1580 FORMAT (1X,15A4,/)
C
      WRITE (2,1551) NOPTS, AVE, SDEV
 1551 FORMAT (/, ' NOPTS =',I5,/,' MEANS =',F10.4,/,
     1 ' STANDARD DEVIATION =',F10.4,///)
      WRITE (3,1551) NOPTS, AVE, SDEV
C
```

```
C  ***** ASSUME INITIAL NUMBER ITERATIONS TO BE THE SQUARE ROOT OF
C          OF THE NUMBER OF POINTS WITH A MAXIMUM OF 19 ITERATIONS     *****
          T=SQRT(PN)
          NITER=T+.5
          IF (NITER .GE. 20) NITER = 19
 1700 CONTINUE
C
C     ***** GIVE USER OPTION OF SPECIFYING NUMBER OF ITERATIONS  *****
          WRITE(3,1710)
 1710 FORMAT (' ENTER NUMBER OF ITERATIONS AS A 2 DIGIT NUMBER OR ',
     1 /,' ENTER BLANKS FOR DEFAULT = SQUARE ROOT OF NUMBER OF POINTS')
          READ(3,1740)NTRIAL
 1740 FORMAT(I2)
          IF(NTRIAL.GT.0)NITER=NTRIAL
 1790 WRITE(2,1800)
 1800 FORMAT (/,'  NUMBER OF  TOTAL NUMBER             STANDARD ',
     1 '   EXTREME      ESD     CRITICAL TEST ',/,
     2          '  POTENTIAL  OBSERVATIONS     MEAN     DEVIATION ',
     3 ' OBSERVATION                VALUES',/,'  OUTLIERS',/)
C
C  ************  BEGIN ITERATIVE PROCESS  **********
          NOITER = NITER
          MAXITR = NITER
 1900 T=0.0
          NP = NOPTS
          DO 1902 I = 1,NP
 1902 M(I) = 0
          CALL ITER (NITER,THETA)
C
C  SKIP LINE AFTER ITERATION DATA IS PRINTED IN SUBROUTINE
          WRITE (2,1111)
 1111 FORMAT (24X)
C
C  CHECK FOR ERROR IN SUBROUTINE
          IF (THEOK .EQ. -9099999.) GO TO 9000
C
C  NO FURTHER TESTING IF NUMBER OF ITERATIONS SPECIFIED
          IF (NTRIAL .GT. 0) GO TO 8000
C
C  CHECK FOR EXPP .GE. THEORY ON LAST ITER
          IF (LSTEGT .EQ. NITER .AND. MAXITR .LT. NOITER) GO TO 8010
          IF (LSTEGT .EQ. NITER) GO TO 2000
C
C  EXPP .LT. THEORY,  BACK UP 1 ITERATION UNLESS NO. OF ITERATIONS = 1.
 1950     MAXITR = MAXITR - 1
          NITER = MAXITR
          IF (NITER .LE. 0) GO TO 8010
          GO TO 1900
C
C  EXPP .GE. THEORY ON LAST ITER, INCREASE NUMBER OF ITERATIONS
 2000 NITER = NITER + 1
C
C  REINITIALIZE CONSTANTS AND BEGIN ITERATIVE CALCULATIONS
          NP = NOPTS
          DO 2002 I = 1,NP
 2002 M(I) = 0
```

```
            CALL ITER (NITER,THETA)
      C
      C  SKIP LINE AFTER ITERATION DATA WRITTEN IN SUBROUTINE
            WRITE (2,1111)
      C
      C  CHECK FOR ERROR IN SUBROUTINE
            IF (THEOK .EQ. -9999999.) GO TO 9000
            IF (EXPPK.LT.THEOK .AND. EXPPK1.LT.THEOK1) GO TO 1950
            IF (EXPPK .LT. THEOK) GO TO 2500
      C
      C  EXPP .GE. THEORY ON KTH, CHECK K+1 TH ITERATION
            IF (EXPPK1 .LT. THEOK1) GO TO 8000
      C
      C  EXPP .GE. THEORY ON KTH, AND K+1 TH ITERATIONS    OR
      C  EXPP(K).LE.THEORY(K) AND EXPP(K+1).GE.THEORY(K+1) ON K+1 TH ITERATION
      2500  IF (NITER+1 .LE. NOPTS/2) GO TO 2000
             GO TO 9200
      C
      C
      C       ***********  GOOD COMPLETION  ***********
      C
      8000  NITER = NITER - 1
      C
      C
      8010  WRITE (2,8011) NITER
      8011  FORMAT ( //,' OUTLIERS = ',I2)
      C
            IF (NITER .EQ. 0) STOP
      8012  WRITE (2,8013) (OUTLIE(I),I=1,NITER)
      8013  FORMAT ( //,' POINTS =  ',7F10.3,/,7F10.3,/,7F10.3)
            STOP
      C
      C
      C    ************  ERROR MESSAGES  ***********
      C
      9000  WRITE (2,9001) NITER
      9001  FORMAT(' FATAL ERROR DURING ITERATION ',I3,', LESS THAN 2 NONZERO'
           1  ,' POINTS')
            STOP
      C
      9200  NOPTS2  = NOPTS/2
            WRITE (2,9201) NITER, NOPTS2
      9201  FORMAT(//,' ITERATION NO. (',I3,') EXCEEDS NO. POINTS/2 (',I3,')')
            STOP
      C
            END
```

```
      SUBROUTINE ITEG (NITER, THETA)
      COMMON ARG,T,AKK,AK1,BKK,BK1,NP,M(1000),D(1000),THEOK,THEOK1,
     1  EXPPK,EXPPK1,LSTEGT,OUTLIE(200)
      DIMENSION THETA(1)
C
C  INITIALIZE CONSTANT FOR LAST ITERATION WHERE EXPP .GE. THEORY
      LSTEGT = 0
C
C  CALCULATE CONSTANTS
      PN = FLOAT(NP)
      ARG=PN/2. - FLOAT(NITER)
      CON1= .000532 * FLOAT(NITER) * FLOAT(NITER)
      CON1= 2.  - EXP(CON1)
      AK1=THETA(1) - THETA(2)*EXP(-THETA(3)) +
     1  THETA(5)*LOG(FLOAT(NITER))
      AKK=THETA(1) - THETA(2)*EXP(-THETA(3)*FLOAT(NITER))
      BK1=THETA(8) - THETA(10)*LOG(FLOAT(NITER))
      BKK=THETA(8) - THETA(9)*LOG(FLOAT(NITER))*CON1
C
C   *********  BEGIN ITERATIONS  *********
      DO 3000 I=1,NITER
C
C  SET QI VALUE
      IF(I.EQ.1)QI=0.0
      IF(I.GT.1.AND.I.LT.NITER)QI=EXP(-EXP(THETA(6)-THETA(7)
     1  * FLOAT(I) ) )
      IF(I.EQ.NITER)QI=1.0
C
C  SET AKI VALUE
      AKI=QI*AKK + (1.-QI)*AK1
      IF(I.EQ.1)AKI=AK1
      IF(I.EQ.NITER)AKI=AKK
C
C  SET T VALUE
      IF(NITER.NE.1)T= (FLOAT(I)-1.) /(FLOAT(NITER)-1.)
C
C  SET BKI VALUE
      T=SQRT(T)
      BKI=T*BKK + (1.-T)*BK1
      IF(I.EQ.1)BKI=BK1
      IF(I.EQ.NITER)BKI=BKK
C
C  FIND SUM OF VALUES AND VALUES SQUARED, COUNT NONZERO VALUES
      SUM1=0.0
      SUM2=0.0
      ICOUNT = 0
      DO 2200 J=1,NP
      IF(M(J).GT.0)GO TO 2200
      ICOUNT = ICOUNT + 1
      SUM1=SUM1+D(J)
      SUM2=SUM2+D(J)*D(J)
 2200 CONTINUE
      IF(ICOUNT.LT.2)GO TO 3100
      COUNT=FLOAT(ICOUNT)
C
C  COMPUTE MEAN, STANDARD DEVIATION
```

```
      AVE1=SUM1/COUNT
      VAR1=(SUM2-SUM1*SUM1/COUNT) / (COUNT-1)
      DEV1=SQRT(VAR1)
C
C  FIND EXTREME VALUE
      BIG=0.0
      DO 2400 J=1,NP
      IF(M(J).GT.0)GO TO 2400
      DIFF=ABS(D(J)-AVE1)
      IF(DIFF.GT.BIG)LOCATE=J
      IF(DIFF.GT.BIG)BIG=DIFF
 2400 CONTINUE
C
C  SAVE EXPERIMENTAL AND THEORICAL VALUES
      IF(DEV1.LT.1.E-20)EXPP=0.0
      IF(DEV1.GT.1.E-20)EXPP=BIG/DEV1
      IF(ARG.GE.1.E-20)ARGLOG=LOG(ARG)
      IF(ARG.LT.1.E-20)ARGLOG=0.0
      THEO=AK1 + BK1*ARGLOG
C
C  SAVE LOCATION OF OUTLIER
      M(LOCATE)=I
      OUTLIE(I) = D(LOCATE)
      WRITE(2,2600) NITER,ICOUNT,AVE1,DEV1, D(LOCATE), EXPP, THEO
 2600 FORMAT(1H ,  3X,I3,9X,I5,3X, 5F11.4)
C
C
C  ********  SAVE EXPP AND THEORY IF LAST OR NEXT TO LAST ITERATION ****
      IF (I .EQ. NITER)   EXPPK1 = EXPP
      IF (I .EQ. NITER)   THEOK1 = THEO
      IF (I .EQ. NITER-1) EXPPK  = EXPP
      IF (I .EQ. NITER-1) THEOK  = THEO
      IF (EXPP .GE. THEO) LSTEGT = I
 3000 CONTINUE
      RETURN
 3100 CONTINUE
      THEOK=-9999999.
      RETURN
      END
```