[redacted]

200666

FATIGUE CRACK GROWTH MODEL
RANDOM2 USER MANUAL

Prepared by :

Lola Boyce, Ph.D., P.E.
Thomas B. Lovelace

APPENDIX 1
of Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for :

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, OH 44135

(NASA-CR-184939) FATIGUE CRACK GROWTH MODEL RANDOM2 USER MANUAL, APPENDIX 1 Annual Report (Texas Univ.) 37 p CSCL 20K N89-23890

Unclas
G3/39 0200666

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989

FATIGUE CRACK GROWTH MODEL
RANDOM2 USER MANUAL

Prepared by :

Lola Boyce, Ph.D., P.E.
Thomas B. Lovelace

APPENDIX 1
of Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for :

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, OH 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989

TABLE OF CONTENTS

1.0 INTRODUCTION

This User Manual documents the FORTRAN program RANDOM2. RANDOM2 is based on fracture mechanics using a probabilistic fatigue crack growth model. It predicts the random lifetime of an engine component to reach a given crack size (see Section 2.0, Theoretical Background).

Included in this Manual are details regarding the theoretical background of RANDOM2, input data instructions and a sample problem illustrating the use of RANDOM2. Appendix A gives information on the physical quantities, their symbols, FORTRAN names, and both SI and U.S. Customary units. Appendix B includes photocopies of the actual computer printout corresponding to the sample problem. Appendices C and D detail the IMSL, Ver. 10 [1], subroutines and functions called by RANDOM2 and a SAS/GRAPH [2] program that can be used to plot both the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.).

2.0 THEORETICAL BACKGROUND

Fatigue crack growth data are usually presented as cycles, N, to reach a particular crack length, a. The initial crack size is a_i. It is generally accepted that under constant amplitude alternating stress, fatigue crack growth can be related to stress intensity through a first order differential equation.[3]

$$da/dN = C(\Delta K)^m \tag{1}$$

where C is a material parameter, m is a material property (often a constant) and ΔK is the stress intensity range. Stress intensity range is given by

$$\Delta K = Y \Delta\sigma \sqrt{\pi a}$$

where Y is a constant dependent upon component and crack geometry and Δσ is the constant amplitude alternating stress. Therefore, equation (1) can be written as

$$da/dN = C(Y \Delta\sigma \sqrt{\pi a})^m$$

or,

$$da/dN = C\, Y^m \Delta\sigma^m \pi^{m/2} a^{m/2}. \tag{2}$$

Equation (2) can be integrated, from the initial crack length, a_i, to the final crack length, a_f, to yield N, the number of cycles. The result is

$$N = \frac{1}{C Y^m \pi^{m/2} \Delta\sigma^m} \left[\frac{a_f^{-m/2+1} - a_i^{-m/2+1}}{-m/2 + 1} \right] \tag{3}$$

Thus, equation (3) gives the "cycles to reach a given crack length."

Metallurgical evidence indicates that casting pores play a significant role in the high-cycle fatigue life of cast nickel base-superalloys, especially at high temperatures.[4] The location and size of these fatigue crack-initiating pores vary greatly from one aerospace propulsion system component to another. This accounts for the large variability in fatigue life and leads to consideration of fatigue crack growth as a random phenomenon.

Fatigue life directly relates to casting pore size, and pore size can be used to determine initial crack size, a_i. Thus, utilizing principles of both probabilistic analysis and fatigue crack growth, a quantitative probabilistic constitutive relationship between fatigue life and fracture mechanics parameters can be developed. Using the "randomized equation" approach, the fatigue crack growth model, given by equation (3) has the following form:

$$N = f(C, m, \Delta\sigma, a_i, a_f, Y) \tag{4}$$

or, in general,

$$N = f(X_i),\ i = 1,....,6, \tag{5}$$

where the X_i are the six independent variables in equations (3) and (4). Equation (3) is "randomized" by assuming the first four variables in equation (4) to be random. Assuming a small crack in a relatively large component leads to assuming $Y = 1.0$, a deterministic value. A deterministic final crack size was chosen since experimental evidence indicated that it was relatively unimportant.[3]

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method. Maximum penalized likelihood generates the p.d.f. estimate using the method of maximum likelihood together with a penalty function to smooth it.[5]

3.0 INPUT DATA

Data input for RANDOM2 is user friendly and easy to manipulate (see, for example, the file entitled NORMAL.INP, in Section 4.0). The first five lines of input have the same format, namely 2E12.4, and the last two lines differ. The last two lines of input have the formats I3,2X,I3,2X,2E12.4,2X,I3 and I3, respectively. A brief line by line description is given along with an example for each line (Note: the ruler is to aid the user in formatting and is not a part of the input). A table listing the physical quantities, their units and symbols is given in Appendix A.

1. Random Number Generator Seed, ISEED, and Sample Size, NTOT

EXAMPLE:

```
123456789012345678901234567890
           1          40
```

2. Material Property, RMM

EXAMPLE:

```
123456789012345678901234567890
    28.0E-01     1.4E-01
```

3. Initial Crack Size (Pore Diameter), RAI

EXAMPLE:

```
123456789012345678901234567890
   300.0E-06    45.0E-06
```

4. Material Property, RCC

EXAMPLE:

```
123456789012345678901234567890
    2.20E-11    0.22E-11
```

5. Stress Range, DELSIG

EXAMPLE:

```
123456789012345678901234567890
     6.2E+02     6.2E+01
```

6. The DESPL [1] parameters are NODE, INIT, ALPHA, EPS, MAXIT and are entered in that order as follows:

EXAMPLE:

```
1234567890123456789012345678901234567890
 21     0         50.0E-01     10.0E-05   30
```

7. The DESPL parameter, IOPT, is entered as follows:

EXAMPLE:

```
1234567890
  2
```

4.0 SAMPLE PROBLEM FOR RANDOM2

The objective of this program is to predict the random lifetime, to reach a given crack size for an engine component. The theory is based on fracture mechanics, using a probabilistic fatigue crack growth model (see Section 2.0, Theoretical Background). RANDOM2 input parameters are given in Table A1.1. Note that the first four parameters are random. Their means and standard deviations are input by the user. The last two parameters, A_f and Y, are deterministic and are fixed internally by the program. They are equal to the values shown in Table A1.1.

Table A1.1 RANDOM2 sample problem input (SI units)

FORTRAN Name	Distribution Type	Mean	Standard Deviation (Value)	Standard Deviation (% of Mean)
RMM	normal	28.0E-01	1.4E-01	(5%)
AI	lognormal	300.0E-06	45.0E-06	(15%)
RCC	lognormal	2.20E-11	0.22E-11	(10%)
DELSIG	lognormal	6.2E+02	6.2E+01	(10%)
AF	N/A	2.0E-03	N/A	
YY	N/A	1.0	N/A	

The input is entered in the following format in a file entitled NORMAL.INP.

```
1234567890123456789012345678901234567890
          1          40
    28.0E-01      1.4E-01
   300.0E-06     45.0E-06
    2.20E-11     0.22E-11
     6.2E+02      6.2E+01
 21    0       50.0E-01     10.0E-05    30
  2
```

Execution of RANDOM2 (source code entitled NR2.FOR) produces an output file entitled RANDM22 giving intermediate results (see Appendix B). Execution also produces the plotfiles OUT1 and OUT2 (see Appendix B). These files are used to plot the X and Y axes of the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.), respectively, generated by RANDOM2. The plots are drawn from the plotfiles by the SAS/GRAPH graphing program (see Appendix C). These plots for the sample problem are shown in Figures A1.1 and A1.2.

This same sample problem has been reported in Boyce and Chamis.[6] There, however, it utilized U.S. Customary units and an older version of RANDOM2 (IMSL Version 9.2 subroutines).

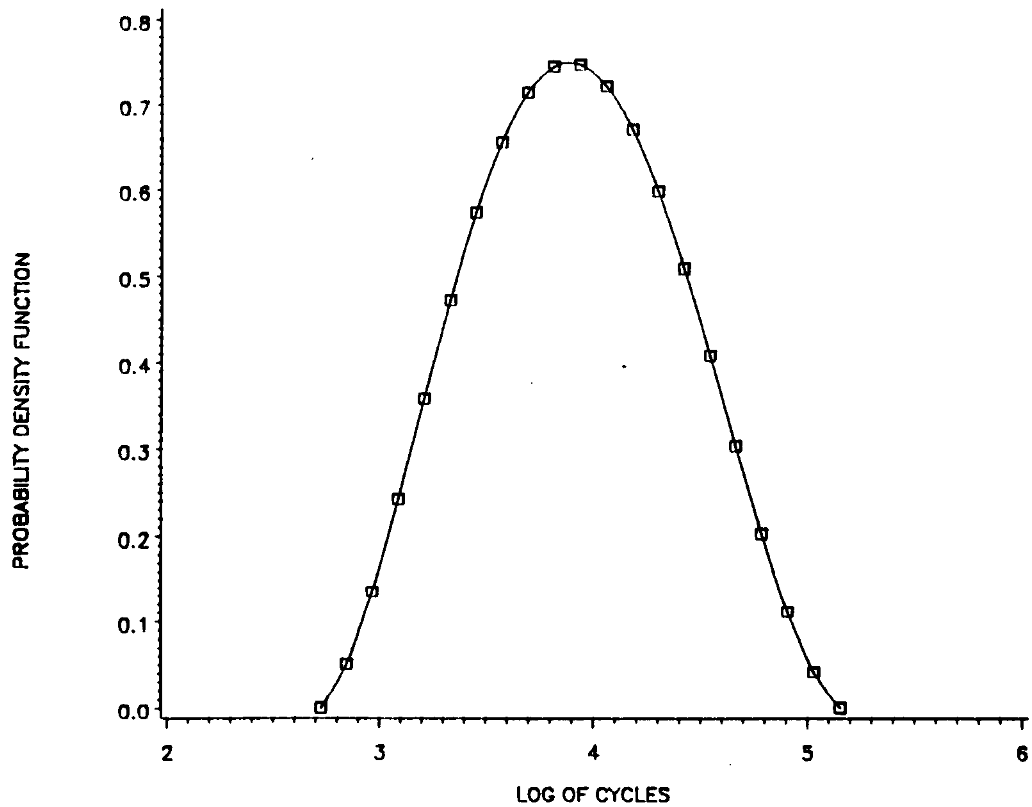

Fig. A1.1 p.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.

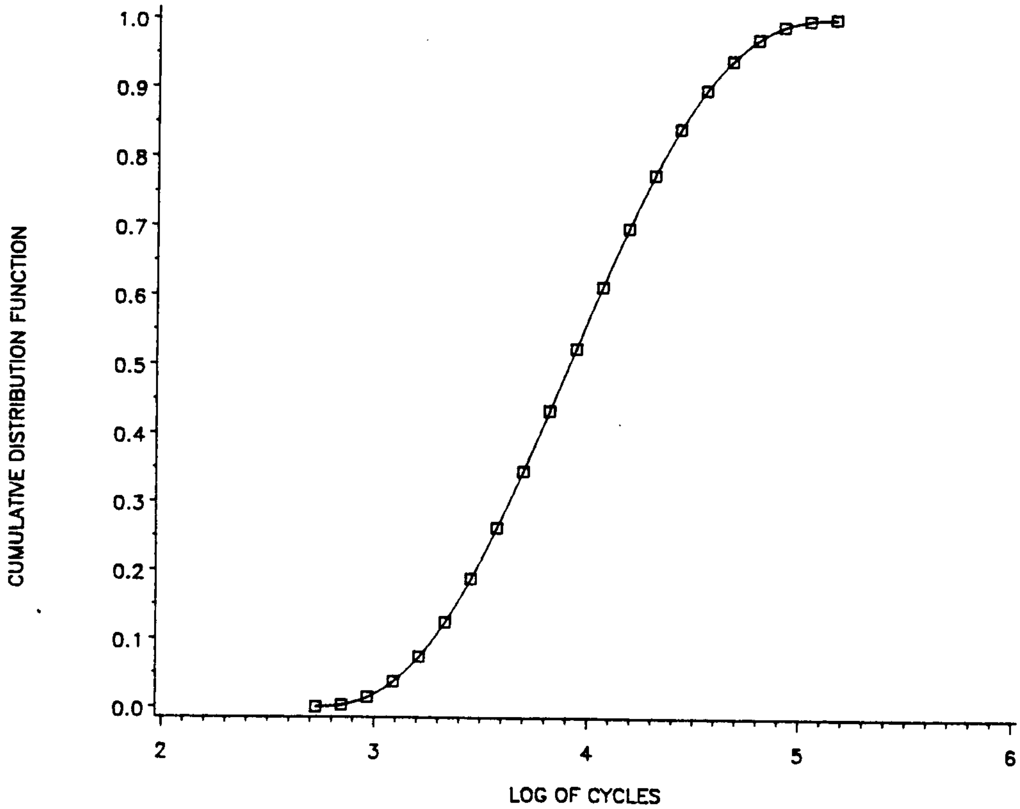

Fig. A1.2 c.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.

5.0 REFERENCES

[1] IMSL, "STAT/LIBRARY, FORTRAN Subroutines for Statistical Analysis", Houston, Texas, 1987.

[2] SAS Institute, Inc. SAS/GRAPH User's Guide, Version 5 Edition, Cary, NC: SAS Institute, Inc., 1985, p. 596.

[3] Kozin, F. and Bogdanoff, J.K., "A Critical Analysis of Some Probabilistic Models of Fatigue Crack Growth," Engineering Fracture Mechanics, Vol. 14, 1981, pp. 55-89.

[4] Hoffeler, W., "High-Cycle Fatigue-Life of the Cast Nickel Base-Superalloys in 738 LC and IN 939," Metallurgical Transactions A, Vol. 13A, July, 1982, pp. 1245-1255.

[5] Scott, D.W., "Nonparametric Probability Density Estimation by Optimization Theoretic Techniques," NASA CR-147763, April, 1976.

[6] Boyce, L. and Chamis, C.C., "Probabilistic Constitutive Relations for Cyclic Material Strength Models," Proceedings, 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, 1988.

6.0 APPENDIX A

PHYSICAL QUANTITIES, SYMBOLS, AND UNITS

The physical quantities, their symbols, and units for the fatigue crack growth model are given in the following table.

Table A1.2 Physical quantities, symbols, and units for fatigue crack growth model for RANDOM2

Physical Quantity	Theory Symbol	FORTRAN Name	Units SI	Units U.S.
Material Property	m	RMM	m/cycle/M Pa m	in/cycle/ksi in
Initial Crack Size	A_i	RAI	m	in
Material Property	C	RCC	m/cycle	in/cycle
Alternating Stress	$\Delta\sigma$	DELSIG	M Pa	ksi
Final Crack Size	A_f	AF	m	in
Geometry Dependent Constant	Y	YY	(dimensionless)	

7.0 APPENDIX B

SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUT FILES

```
DDDDDDDDDD    3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333    DDDDDDDD
DDDDDDDDDD    333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333    DDDDDDDD
DDDDDDDDDD    3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333    DDDDDDDD

                              DDDD   EEEEE   CCCC  N    N  EEEEE  TTTTT
                              D   D  E      C      N    N  E        T
                              D   D  E      C      NN   N  E        T
                              D   D  EEEE   C      N N  N  EEEE     T
                              D   D  E      C      N  NN   E        T
                              D   D  E      C      N    N  E        T
                              DDDD   EEEEE   CCCC  N    N  EEEEE    T

                                NN       NN  RRRRRRRR       222222
                                NN       NN  RRRRRRRR       222222
                                NN       NN  RR      RR  22        22
                                NN       NN  RR      RR  22        22
                                NNNN     NN  RR      RR            22
                                NNNN     NN  RR      RR            22
                                NN  NN   NN  RRRRRRRR            22
                                NN  NN   NN  RRRRRRRR            22
                                NN    NNNN   RR  RR            22
                                NN    NNNN   RR  RR            22
                                NN       NN  RR    RR       22
                                NN       NN  RR    RR       22
                                NN       NN  RR      RR  2222222222
                                NN       NN  RR      RR  2222222222

                          FFFFFFFFFF    000000      RRRRRRRR      ;;;;    77777777
                          FFFFFFFFFF    000000      RRRRRRRR      ;;;;    77777777
                          FF          00      00    RR      RR    ;;;;          77
                          FF          00      00    RR      RR    ;;;;          77
                          FF          00      00    RR      RR                  77
                          FF          00      00    RR      RR                  77
                          FFFFFFFF    00      00    RRRRRRRR      ;;;;        77
                          FFFFFFFF    00      00    RRRRRRRR      ;;;;        77
                          FF          00      00    RR  RR        ;;;;      77
                          FF          00      00    RR  RR        ;;;;      77
                  ....    FF          00      00    RR    RR        ;;    77
                  ....    FF          00      00    RR    RR        ;;    77
                  ....    FF            000000      RR      RR    ;;    77
                  ....    FF            000000      RR      RR    ;;    77
```

File _DUA0:[]NR2.FOR;7 (648,66,0), last revised on 29-NOV-1988 11:19, is a 43 block sequential file owned by UIC [DECNET]. The records are variable length with implied (CR) carriage control. The longest record is 72 bytes.

Job NR2 (1814) queued to TERM$LA120A on 29-NOV-1988 11:21 by user DECNET, UIC [DECNET], under account DECNET at priority 100, starte on printer LTA4: on 29-NOV-1988 11:21 from queue TERM$LA120A.


```
DDDDDDDDDD    3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333    DDDDDDDD
DDDDDDDDDD    333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333    DDDDDDDD
DDDDDDDDDD    3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333    DDDDDDDD
```

```
JOB,JN=COMPL,US=USAQ530,RT=60,T=30,MFL=3000000.
ACCOUNT,UPW=LOLAB.
lib,dn=imslib.
CFT77,LIST.
REWIND,DN=$BLD.
BUILD,I=0,OBL=0.
SAVE,DN=$NBL,PDN=NR2BLD,ID=SMBOYCE.
DELETE,PDN=NR2BLD,ID=SMBOYCE,ED=-1.
/EOF
C PARIS-ERDOGAN FATIGUE CRACK GROWTH EQUATION
C RANDOMIZED AND APPLIED TO CRACK LENGTH
C WHERE INITIAL CRACK SIZE IS RANDOM PORE DIAMETER
      INTEGER ISEED,NTOT,M,INIT,NMISS,MAXIT,NODE
      REAL XM,XS,YM,YS,EPS,P,RWKSP(9999),ALPHA
      COMMON/WORKSP/RWKSP
      DIMENSION RMM(10000),RAI(10000),RCC(10000)
      DIMENSION DELSIG(10000),BNDSX(1000)
      DIMENSION XNF(10000),C(10000)
      DIMENSION STAT(9999),DENS(1000),DISTX(1000)
      DIMENSION BNDS(10000),BB(1000),FF(1000)
      EXTERNAL RNLNL,RNSET,RNNOR,DESPL,IWKIN
 1001 FORMAT(5E12.4)
 1002 FORMAT(I12,I12)
 1003 FORMAT(I4,I4)
 1004 FORMAT(I4)
 1009 FORMAT(I3,2X,I3,2X,2E12.4,2X,I3)
 1010 FORMAT(I3)
 1011 FORMAT(2E12.4)
C NORMAL MATERIAL PROPERTY, M
      READ (5,1002)ISEED,NTOT
      WRITE(6,1002)ISEED,NTOT
      READ (5,1011)YM,YS
      WRITE(6,1011)YM,YS
C       YS=0.14
C       YM=2.8
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,RMM)
      DO 102 I=1,NTOT
      RMM(I)=YS*RMM(I)+YM
  102 CONTINUE
      WRITE(6,2019)
 2019 FORMAT(' MATERIAL PROPERTY, M')
      WRITE( 6,1001) (RMM(I),I=1,NTOT)
C LOGNORMAL INITIAL CRACK SIZE (PORE DIAMETER),AI
      WRITE(6,1002) ISEED,NTOT
      READ (5,1011) XM,XS
      WRITE(6,1011) XM,XS
C       XM  = 300.0E-06
C       XS  = 45.0E-06
      YS = SQRT( LOG(1.0+(XS/XM)**2) )
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,RAI)
      WRITE(6,2020)
 2020 FORMAT(' INITIAL CRACK SIZE, AI')
      WRITE( 6,1001) (RAI(I),I=1,NTOT)
C LOGNORMAL MATERIAL PROPERTY, C
      WRITE(6,1002) ISEED,NTOT
      READ (5,1011) XM,XS
      WRITE(6,1011) XM,XS
C       XM  = 2.20E-11
C       XS  = 0.220E-11
      YS = SQRT( LOG(1.0+(XS/XM)**2) )
      YM = LOG(XM) - 0.5*YS**2
```

ORIGINAL PAGE IS OF POOR QUALITY

```fortran
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,RCC)
      WRITE(6,2021)
 2021 FORMAT(' MATERIAL PROPERTY, C')
      WRITE( 6,1001) (RCC(I),I=1,NTOT)
C LOGNORMAL STRESS RANGE, DELSIG
      WRITE(6,1002)ISEED,NTOT
      READ (5,1011) XM,XS
      WRITE(6,1011) XM,XS
C       XS = 0.62E+02
C       XM = 6.2E+02
      YS = SQRT(LOG(1.0+(XS/XM)**2))
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,DELSIG)
      WRITE (6,2022)
 2022 FORMAT (' STRESS RANGE, DELSIG')
      WRITE (6,1001)(DELSIG(I),I=1,NTOT)
C DEFINE DETERMINISTIC PARAMETERS
C PI
      PI = 3.14159265358979323846264 33
C COMPONENT AND CRACK SHAPE PARAMETER, YY
      YY=1.0
C FINAL CRACK SIZE, AF
      AF=2.0E-03
C CALCULATE CYCLES TO REACH CRACK SIZE 2.0E-03M
      DO 101 I=1,NTOT
      XNF1=1.0/(RCC(I)*YY**RMM(I)*PI**(RMM(I)/2.)*DELSIG(I)**
     1RMM(I))
      XNF2=((AF**(1.-RMM(I)/2.)-RAI(I)**(1.-RMM(I)/2.))/
     1(1.-RMM(I)/2.))
      XNF(I)=XNF1*XNF2
C CALCULATE LOG OF CYCLES TO REACH CRACK SIZE 2.0E-03M
      XNF(I)=ALOG10(XNF(I))
  101 CONTINUE
      WRITE(6,2023)
 2023 FORMAT (' LOG OF CYCLES TO REACH CRACK SIZE=2.0E-03M,',/,
     1'GIVEN STRESS MEAN AMPLITUDE=6.2E+02MPA')
      WRITE(6,1001)(XNF(I),I=1,NTOT)
C SORT LOG OF CYCLES
      CALL SORT(XNF,NTOT)
      WRITE (6,2024)
 2024 FORMAT (' SORTED LOG OF CYCLES')
      WRITE (6,1001)(XNF(I),I=1,NTOT)
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C CALCULATE PDF OF LOG OF CURRENT CYCLES,LOG XNF
      READ(5,1009)NODE,INIT,ALPHA,EPS,MAXIT
      WRITE(6,985)
  985 FORMAT(' DESPL PARAMETERS')
      WRITE(6,1009)NODE,INIT,ALPHA,EPS,MAXIT
      BNDS(1)=XNF(1) - 0.05*XNF(1)
      BNDS(2)=XNF(NTOT) + 0.05*XNF(NTOT)
      WRITE(6,979)BNDS(1),BNDS(2)
  979 FORMAT('BNDS(1),BNDS(2)=',E12.4,1X,E12.4)
      CALL DESPL(NTOT,XNF,NODE,BNDS,INIT,ALPHA,MAXIT,EPS,DENS,STAT,
     1NMISS)
      WRITE(6,980)
  980 FORMAT('PDF OF LOG OF CURRENT CYCLES,LOG XNF,Y AXIS OF PDF PLOT')
      WRITE(6,1001)(DENS(I),I=1,NODE)
      WRITE(6,981)
  981 FORMAT(' OUTPUT STATISTICS')
      WRITE(6,1001)(STAT(I),I=1,4)
      WRITE(6,982)
  982 FORMAT(' NUMBER OF MISSING VALUES')
```

```
      WRITE(6,1010)NMISS
C
C CALCULATE WINDOW WIDTH, HH
C
      HH=(BNDS(2)-BNDS(1))/(NODE-1)
C
C CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED;
C ALSO CALLED 'NODE' VALUES
C
      DO 6001,I=1,NODE-2
      BNDS(I+2)=BNDS(1) + (I*HH)
 6001 CONTINUE
      WRITE(6,983)
  983 FORMAT(' LOG OF CURRENT CYCLES, LOG XNF')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C
C REORDER BNDS FOR PLOTTING
C
      SAVE1 = BNDS(2)
      SAVE2 = BNDS(NODE)
      BNDS(NODE)=BNDS(2)
      DO 6002, I=1, NODE-2
      BNDS(I+1)=BNDS(I+2)
 6002 CONTINUE
      BNDS(NODE-1)=SAVE2
      BNDS(NODE)=SAVE1
      WRITE(6,984)
  984 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG,XNF,
     1X AXIS PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES,
C   LOG XNF TO PLOT FILES
      WRITE(34,990)
  990 FORMAT(' (E12.4,1X,E12.4)')
      WRITE(34,991)(BNDS(J),DENS(J),J=1,NODE)
  991 FORMAT(E12.4,1X,E12.4)
C
C CALCULATE CDF OF LOG OF CURRENT CYCLES
C
      READ(5,1010)IOPT
      WRITE(6,992)
  992 FORMAT(' GCDF PARAMETERS')
      WRITE(6,1010)IOPT
      XO=BNDS(1)
      DO 6003,I=1,NODE
      P=GCDF(XO,IOPT,NODE,BNDS,DENS)
      BNDSX(I)=XO
      XO=XO+HH
      DISTX(I)=P
 6003 CONTINUE
      WRITE(6,994)
  994 FORMAT(' CDF OF LOG OF CURRENT CYCLES, LOG XNF,
     1Y AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(DISTX(I),I=1,NODE)
C
      WRITE(6,993)
  993 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG XNF,
     1X AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
      WRITE(6,1001)(BNDSX(I),I=1,NODE)
C
C WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT
C TO THE PLOT FILES
C
```

```
      WRITE(35,990)
      WRITE(35,991)(BNDS(J),DISTX(J),J=1,NODE)
      STOP
      END
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      SUBROUTINE SORT (Y,N)
      DIMENSION Y(10000)
C Y IS THE ARRAY TO BE SORTED
C AT COMPLETION Y(1) IS SMALLEST VALUE
C AT COMPLETION Y(N) IS LARGEST VALUE
      N1 = N - 1
      DO 1 I=1,N1
      J = I + 1
      DO 2 K=J,N
      IF (Y(I).LT.Y(K))GO TO 2
      TEMP = Y(I)
      Y(I) = Y(K)
      Y(K) = TEMP
    2 CONTINUE
    1 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
C  IMSL Name:  D3SPL/DD3SPL (Single/Double precision version)
C
C  Computer:   IBM/SINGLE
C
C  Revised:    November 1, 1985
C
C  Purpose:    Nonparametric probability density function estimation
C              estimation by the penalized likelihood method.
C
C  Usage:      CALL D3SPL (NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS,
C                          DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B,
C                          IPVT, WK2)
C
C  Arguments:
C     NOBS   - Number of observations.  (Input)
C     X      - Vector of length NOBS containing the random sample of
C              responses.  (Input)
C     NODE   - Number of mesh nodes for the discrete pdf estimate.
C              (Input)
C     BNDS   - Vector of length 2 containing the minimum and maximum
C              values for X(i) in BNDS(1) and BNDS(2), respectively.
C              (Input)
C     INIT   - Initialization option.  (Input)
C     ALPHA  - Positive penalty weighting factor which controls the
C              smoothness of the estimate.  (Input)
C     MAXIT  - Maximum number of iterations allowed in the iterative
C              procedure.  (Input)
C     EPS    - Convergence criterion.  (Input)
C     DENS   - Vector of length NODE containing the estimated values of
C              the discrete pdf at the NODE equally spaced mesh nodes.
C              (Input/output if INIT=1, Output otherwise)
C     STAT   - Vector of length 4 containing out statistics.  (Output)
C              STAT(1) and STAT(2) contain the log-likelihood and the
C              log-penalty terms, respectively.  STAT(3) and STAT(4)
C              contain the estimated mean and variance for the
C              estimated density.
C     HESS   - Seven by NODE-2 hessian matrix (and its factorization).
C              (Output)
C     LDHESS - Leading dimension of HESS exactly as specified in the
C              dimension statement in the calling program.  (Input)
C     ILOHI  - NODE by 2 matrix containing the indices for the risk set
```

16

```
C           at each node value.  (Output)
C    DENEST - NODE by 3matrix containing the gradient vector, among
C             other quantities.  (Output)
C    B      - Vector of length NODE containing the NODE values.
C             (Output)
C    IPVT   - Pivot vector of length NODE-2.  (Output)
C    WK2    - Work vector of length NODE-2.  (Output)
C
C  Chapter:    STAT/LIBRARY Density and Hazard Estimation
C
C  Copyright:  1985 by IMSL, Inc.  All Rights Reserved.
C
C  Warranty:   IMSL warrants only that IMSL testing has been applied
C              to this code.  No other warranty, expressed or implied,
C              is applicable.
C
C-----------------------------------------------------------------------
C
      SUBROUTINE D3SPL (NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS,
     &                  DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B,
     &                  IPVT, WK2)
C                                  SPECIFICATIONS FOR ARGUMENTS
      INTEGER    NOBS, NODE, INIT, MAXIT, LDHESS, ILOHI(NODE,*),
     &           IPVT(*)
      REAL       ALPHA, EPS, X(*), BNDS(2), DENS(*), STAT(4),
     &           HESS(LDHESS,*), DENEST(NODE,*), B(*), WK2(*)
C                                  SPECIFICATIONS FOR LOCAL VARIABLES
      INTEGER    I, IMPTR, IPTR, ITER, K, KM1, KM2, KP1, KP2, M, MOLD,
     &           NER, NOB1
      REAL       BK, BKM1, BSMALL, CK, CKM1, CKM2, CKMCM1, CKP1, CKP2,
     &           CONS, EPS1, FACTOR, FK, FKM1, FKM2, FKP1, H, H2, H3,
     &           SUM, TEMP, WK(4)
      DOUBLE PRECISION SUM1, SUM2, SUM3
C                                  SPECIFICATIONS FOR SAVE VARIABLES
      INTEGER    MINCR(8)
      SAVE       MINCR
C                                  SPECIFICATIONS FOR INTRINSICS
C     intrinsic  alog,amax1,max0,min0,mod,sqrt
      INTRINSIC  ALOG, AMAX1, MAX0, MIN0, MOD, SQRT
      INTEGER    MAX0, MIN0, MOD
      REAL       ALOG, AMAX1, SQRT
C                                  SPECIFICATIONS FOR SUBROUTINES
      EXTERNAL   E1MES, E1POP, E1PSH, E1STI, E1STR, SADD, SAXPY,
     &           SCOPY, SHPROD, SSCAL, D2SPT, L2TRB, LFSRB
C                                  SPECIFICATIONS FOR FUNCTIONS
      EXTERNAL   ISMIN, N1RCD, SDOT, SNRM2, SSUM
      INTEGER    ISMIN, N1RCD
      REAL       SDOT, SNRM2, SSUM
C
      DATA MINCR/5, 9, 17, 33, 65, 129, 253, 100001/
C
      CALL E1PSH ('D3SPL ')
C                                  Error checks
      NER = 1
      IF (NOBS .LT. 1) THEN
         CALL E1MES (5, 1, 'After removing all missing (NaN, not a '//
     &               'number) values from X there are no valid '//
     &               'observations.  At least one valid observation '//
     &               'is necessary.')
      END IF
      IF (NODE .LE. 4) THEN
         CALL E1STI (1, NODE)
         CALL E1MES (5, 2, 'NODE = %(I1).  The number of mesh '//
     &               'nodes, NODE, must be an odd integer greater '//
```

```
     &                 'than 4.')
      ELSE IF (MOD(NODE,2) .EQ. 0) THEN
         CALL E1STI (1, NODE)
         CALL E1MES (5, 3, 'NODE = %(I1) must be an odd integer '//
     &                 'greater than 4.')
      END IF
      IF (ALPHA .LE. 0.0) THEN
         CALL E1STR (1, ALPHA)
         CALL E1MES (5, 4, 'ALPHA = %(R1).  The penalty weighting '//
     &                 'factor which controls smoothness, ALPHA, must '//
     &                 'be greater than 0.')
      END IF
      IF (MAXIT .LE. 0.0) THEN
         CALL E1STI (1, MAXIT)
         CALL E1MES (5, 5, 'MAXIT = %(I1).  The maximum number '//
     &                 'of iterations, MAXIT, must be greater than 0.')
      END IF
      IF (BNDS(1) .GT. BNDS(2)) THEN
         CALL E1STR (1, BNDS(1))
         CALL E1STR (2, BNDS(2))
         CALL E1MES (5, 6, 'BNDS(1) = %(R1) and BNDS(2) = '//
     &                 '%(R2).  The minimum value for X, BNDS(1), must '//
     &                 'be less than or equal to the maximum value for '//
     &                 'X, BNDS(2).')
      END IF
      IF (INIT .NE. 0) THEN
         IF (DENS(1).NE.0 .OR. DENS(NODE).NE.0) THEN
            CALL E1STR (1, DENS(1))
            CALL E1STR (2, DENS(NODE))
            CALL E1STI (1, NODE)
            CALL E1MES (5, 7, 'DENS(1) = %(R1) and DENS(NODE=%(I1)) '//
     &                    '= %(R2).  The beginning and ending initial '//
     &                    'estimates of the density must be zero.')
         END IF
         IF (DENS(ISMIN(NODE,DENS,1)) .LT. 0) THEN
            CALL E1MES (5, 8, 'The initial estimates of the '//
     &                    'density, DENS, must be greater than or '//
     &                    'equal to 0.')
         END IF
      END IF
      NOB1 = 0
      DO 10  I=1, NOBS
         IF (X(I).LT.BNDS(1) .OR. X(I).GT.BNDS(2)) THEN
            NOB1 = NOB1 + 1
         END IF
   10 CONTINUE
      IF (NOB1 .EQ. NOBS) THEN
         CALL E1MES (5, 9, 'All elements in X lie outside the '//
     &                 'interval BNDS(1) to BNDS(2).  At least one '//
     &                 'element of X must lie in this interval.')
      END IF
      IF (EPS .LE. 0.0) THEN
         EPS1 = 1.0E-4
      ELSE
         EPS1 = EPS
      END IF
      IF (N1RCD(0) .NE. 0) GO TO 9000
C                                  Initialization
      IMPTR = 0
C                                  Set initial densities
      IF (INIT .EQ. 0) THEN
         DENS(1) = 0.0
         DENS(2) = 2.0/(BNDS(2)-BNDS(1))
         DENS(3) = 0.0
```

```
         M = 3
      ELSE
         M = NODE
      END IF
C                                           Refine mesh
   20 IF (INIT .EQ. 0) THEN
         MOLD = M
         IMPTR = IMPTR + 1
         M = MIN0(NODE,MINCR(IMPTR))
      END IF
C                                           Get mesh interval width
      H = (BNDS(2)-BNDS(1))/(M-1)
      H2 = H*H
      H3 = H2*H
C                                           Make initial DENS integrate to 1.
      IF (INIT .NE. 0) THEN
         CALL SSCAL (NODE, 1.0/(H*SSUM(NODE,DENS,1)), DENS, 1)
      END IF
C                                           Set mesh nodes
      B(1) = BNDS(1)
      DO 30  I=2, M
         B(I) = B(I-1) + H
   30 CONTINUE
C                                           Set B indices for interpolating X
      IPTR = 0
   40 IPTR = IPTR + 1
      IF (X(IPTR) .LT. BNDS(1)) GO TO 40
      DO 60  K=1, M - 1
         ILOHI(K,1) = IPTR
         ILOHI(K,2) = IPTR - 1
         IF (IPTR .LE. NOBS) THEN
   50       IF (X(IPTR) .LT. B(K+1)) THEN
               ILOHI(K,2) = ILOHI(K,2) + 1
               IPTR = IPTR + 1
               IF (IPTR .LE. NOBS) GO TO 50
            END IF
         END IF
   60 CONTINUE
   70 FACTOR = 2.0*ALPHA/H3
C                                           Initialize mesh node densities
      IF (INIT .EQ. 0) THEN
C                                           Via DESPT
         CALL D2SPT (M-2, B(2), 1, MOLD, BNDS, DENS, DENEST, WK, WK,
     &               WK)
         TEMP = 1.0/(M*M*H)
         DO 80  I=2, M - 1
            DENS(I) = AMAX1(TEMP,SQRT(DENEST(I-1,1)))
   80    CONTINUE
      ELSE
C                                           Via the initial estimates
         DO 90  I=2, M - 1
            DENS(I) = SQRT(DENS(I))
   90    CONTINUE
      END IF
      DENS(M) = 0.0
C                                           Maximize
      DO 140  ITER=1, MAXIT
C                                           Get Hessian - Lagrangian
         HESS(1,1) = 0.0
         HESS(1,2) = 0.0
         HESS(2,1) = 0.0
         BSMALL = 0.0
         SUM = 0.0
C                                           CK** are true estimates = FK**2
```

```
      DO 120  K=2, M - 1
         KM1 = K - 1
         KM2 = MAX0(1,K-2)
         KP1 = K + 1
         KP2 = MIN0(M,K+2)
         FK = DENS(K)
         FKM1 = DENS(KM1)
         FKM2 = DENS(KM2)
         CKM2 = FKM2**2
         CKM1 = FKM1**2
         CK = FK**2
         CKP1 = DENS(KP1)**2
         CKP2 = DENS(KP2)**2
         BK = B(K)
         BKM1 = B(KM1)
         SUM = SUM + CK
         IF (K .GE. 4) HESS(1,KM1) = 4.0*FK*FKM2*FACTOR
         SUM1 = 0.0D0
         SUM2 = 0.0D0
         SUM3 = 0.0D0
         DO 100  I=ILOHI(K,1), ILOHI(K,2)
            TEMP = (X(I)-BK)/H
            CONS = (1.0-TEMP)/(CK+(CKP1-CK)*TEMP)
            SUM1 = SUM1 - CONS
            SUM2 = SUM2 + CONS*CONS
  100    CONTINUE
         CKMCM1 = CK - CKM1
         DO 110  I=ILOHI(KM1,1), ILOHI(KM1,2)
            CONS = (X(I)-BKM1)/H
            TEMP = CKM1 + CKMCM1*CONS
            SUM1 = SUM1 - CONS/TEMP
            TEMP = TEMP*TEMP
            SUM2 = SUM2 + (CONS*CONS)/TEMP
            SUM3 = SUM3 + CONS*(1.0-CONS)/TEMP
  110    CONTINUE
         TEMP = FACTOR*(CKM2+CKP2-4.0*(CKM1+CKP1)+6.0*CK) + SUM1
         TEMP = 2.0*TEMP
         BSMALL = BSMALL + 2.0*CK*TEMP
         HESS(3,KM1) = TEMP + 4.0*CK*(6.0*FACTOR+SUM2)
         IF (K .NE. 2) HESS(2,KM1) = 4.0*FK*FKM1*(-4.0*FACTOR+SUM3)
         DENEST(KM1,1) = FK*TEMP
         DENEST(KM1,2) = -2.0*FK
  120 CONTINUE
      BSMALL = 1.0/H - SUM + BSMALL
C                                  Save portion of DENEST
      CALL SCOPY (M-2, DENEST(1,2), 1, DENEST(1,3), 1)
C                                  Finish with the hessian
      CALL SADD (M-2, -BSMALL/(2.0*SUM), HESS(3,1), LDHESS)
C                                  Fill out symmetric band structure
      CALL SCOPY (M-4, HESS(1,3), LDHESS, HESS(5,1), LDHESS)
      HESS(5,M-3) = 0.0
      HESS(5,M-2) = 0.0
      CALL SCOPY (M-3, HESS(2,2), LDHESS, HESS(4,1), LDHESS)
      HESS(4,M-2) = 0.0
C                                  Solve symmetric band linear system
      CALL L2TRB (M-2, HESS, LDHESS, 2, 2, HESS, LDHESS, IPVT, WK2)
      CALL LFSRB (M-2, HESS, LDHESS, 2, 2, IPVT, DENEST, 1, DENEST)
      CALL LFSRB (M-2, HESS, LDHESS, 2, 2, IPVT, DENEST(1,2), 1,
     &            DENEST(1,2))
      IF (N1RCD(1) .NE. 0) GO TO 9000
C                                  Compute the constant
      CONS = SDOT(M-2,DENEST(1,3),1,DENEST(1,2),1)
      CONS = (1.0/H-SUM-SDOT(M-2,DENEST(1,3),1,DENEST(1,1),1))/CONS
C                                  Update the gradient
```

20

```
      CALL SAXPY (M-2, CONS, DENEST(1,2), 1, DENEST(1,1), 1)
C                                  Parameter updates
      CALL SAXPY (M-2, -1.0, DENEST(1,1), 1, DENS(2), 1)
C                                  Check the convergence criterion
      TEMP = SNRM2(M-2,DENS(2),1)
      IF (SNRM2(M-2,DENEST,1) .LT. EPS1*TEMP) GO TO 150
C                                  Ad hoc projection to plus quadrant
      TEMP = TEMP*1.0E-4/SQRT(M-2.0)
      DO 130  I=2, M - 1
         DENS(I) = AMAX1(TEMP,DENS(I))
  130 CONTINUE
  140 CONTINUE
      CALL E1STI (1, MAXIT)
      CALL E1MES (3, 1, 'The maximum number of iterations '//
     &            '(MAXIT=%(I1)) was exceeded.')
C                                  Replace DENS(*) with squares
  150 CALL SHPROD (M-2, DENS(2), 1, DENS(2), 1, DENS(2), 1)
      IF (M .NE. NODE) GO TO 20
C                                  Evaluate log likelihood and penalty
      SUM1 = 0.0
C                                  Penalty
      DO 160  K=1, M
         KM1 = MAX0(K-1,1)
         KP1 = MIN0(K+1,M)
         SUM1 = SUM1 + (DENS(KM1)-2.0*DENS(K)+DENS(KP1))**2
  160 CONTINUE
      STAT(2) = -0.5*FACTOR*SUM1
      SUM2 = 0.0
C                                  Log-likelihood
      DO 170  I=1, NOBS
         IF (X(I).GE.BNDS(1) .AND. X(I).LE.BNDS(2)) THEN
            CALL D2SPT (1, X(I), 1, NODE, BNDS, DENS, DENEST, WK, WK,
     &                  WK)
            SUM2 = SUM2 + ALOG(DENEST(1,1))
         END IF
  170 CONTINUE
      STAT(1) = SUM2
C                                  Evaluate M.L.P.E. mean and variance
      SUM1 = 0.0
      SUM2 = 0.0
      DO 180  K=1, M - 1
         FK = DENS(K)
         FKP1 = DENS(K+1)
         BK = B(K)
         CONS = FK + FKP1
         TEMP = CONS + FKP1
         SUM1 = SUM1 + H2*TEMP/6.0 + 0.5*H*BK*CONS
         SUM2 = SUM2 + H3*(TEMP+FKP1)/12.0 + H2*BK*TEMP/3.0 +
     &          0.5*H*BK*BK*CONS
  180 CONTINUE
      STAT(3) = SUM1
      STAT(4) = SUM2 - SUM1*SUM1
C                                  Exit section
 9000 CALL E1POP ('D3SPL ')
      RETURN
      END
/EOF
```

```
EEEEEEEEEE    4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
EEEEEEEEEE    44444444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  44444444444444444444444444    EEEEEEEE
EEEEEEEEEE    4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE

                              DDDD   EEEEE   CCCC  N   N  EEEEE  TTTTT
                              D   D  E      C      N   N  E        T
                              D   D  E      C      NN  N  E        T
                              D   D  EEEE   C      N N N  EEEE     T
                              D   D  E      C      N  NN  E        T
                              D   D  E      C      N   N  E        T
                              DDDD   EEEEE   CCCC  N   N  EEEEE    T

                    NN      NN    000000    RRRRRRRR    MM      MM    AAAAAA    LL
                    NN      NN    000000    RRRRRRRR    MM      MM    AAAAAA    LL
                    NN      NN  00      00  RR      RR  MMMM  MMMM  AA      AA  LL
                    NN      NN  00      00  RR      RR  MMMM  MMMM  AA      AA  LL
                    NNNN    NN  00      00  RR      RR  MM  MM  MM  AA      AA  LL
                    NNNN    NN  00      00  RR      RR  MM  MM  MM  AA      AA  LL
                    NN  NN  NN  00      00  RRRRRRRR    MM      MM  AA      AA  LL
                    NN  NN  NN  00      00  RRRRRRRR    MM      MM  AA      AA  LL
                    NN    NNNN  00      00  RR  RR      MM      MM  AAAAAAAAAA  LL
                    NN    NNNN  00      00  RR  RR      MM      MM  AAAAAAAAAA  LL
                    NN      NN  00      00  RR    RR    MM      MM  AA      AA  LL
                    NN      NN  00      00  RR    RR    MM      MM  AA      AA  LL
                    NN      NN    000000    RR      RR  MM      MM  AA      AA  LLLLLLLLLL
                    NN      NN    000000    RR      RR  MM      MM  AA      AA  LLLLLLLLLL

                          IIIIII  NN      NN  PPPPPPPP    ;;;;      11      222222
                          IIIIII  NN      NN  PPPPPPPP    ;;;;      11      222222
                            II    NN      NN  PP      PP  ;;;;    1111    22      22
                            II    NN      NN  PP      PP  ;;;;    1111    22      22
                            II    NNNN    NN  PP      PP            11            22
                            II    NNNN    NN  PP      PP            11            22
                            II    NN  NN  NN  PPPPPPPP    ;;;;      11          22
                            II    NN  NN  NN  PPPPPPPP    ;;;;      11          22
                            II    NN    NNNN  PP          ;;;;      11        22
                            II    NN    NNNN  PP          ;;;;      11        22
                  ....      II    NN      NN  PP            ;;      11      22
                  ....      II    NN      NN  PP            ;;      11      22
                  ....    IIIIII  NN      NN  PP          ;;    111111  2222222222
                  ....    IIIIII  NN      NN  PP          ;;    111111  2222222222
```

File _DUA0:[]NORMAL.INP;12 (487,157,0), last revised on 29-NOV-1988 11:22, is a 1 block sequential file owned by UIC [DECNET]. The records are variable length with implied (CR) carriage control. The longest record is 39 bytes.

Job NORMAL (1815) queued to TERM$LA120A on 29-NOV-1988 11:22 by user DECNET, UIC [DECNET], under account DECNET at priority 100, started on printer LTA4: on 29-NOV-1988 11:24 from queue TERM$LA120A.


```
EEEEEEEEEE    4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
EEEEEEEEEE    44444444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  44444444444444444444444444    EEEEEEEE
EEEEEEEEEE    4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
```

```
            1          40
     28.0E-01     1.4E-01
    300.0E-06    45.0E-06
     2.20E-11    0.22E-11
      6.2E+02     6.2E+01
21         0    50.0E-01    10.0E-05   30
 2
```

CCCCCCCCCC 2222222222222222222222222 Digital Equipment Corporation - VAX/VMS Version V4.7 2222222222222222222222222 CCCCCCCC

DECNET

RANDM22

CPR;1

ORIGINAL PAGE IS OF POOR QUALITY

File _DUA0:[]RANDM22.CPR;1 (487,156,0), last revised on 29-NOV-1988 11:19, is a 24 block sequential file owned by UIC [DECNET]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 120 bytes.

Job RANDM22 (1813) queued to TERM$LA120A on 29-NOV-1988 11:19 by user DECNET, UIC [DECNET], under account DECNET at priority 100, started on printer LTA4: on 29-NOV-1988 11:19 from queue TERM$LA120A.

CCCCCCCCCC 2222222222222222222222222 Digital Equipment Corporation - VAX/VMS Version V4.7 2222222222222222222222222 CCCCCCCC

```
         1          40
 0.2800E+01  0.1400E+00
MATERIAL PROPERTY, M
 0.2195E+01  0.2643E+01  0.2897E+01  0.2785E+01  0.2812E+01
 0.2691E+01  0.2566E+01  0.2865E+01  0.2865E+01  0.3012E+01
 0.2759E+01  0.2807E+01  0.2934E+01  0.2546E+01  0.2574E+01
 0.2810E+01  0.2862E+01  0.2461E+01  0.2758E+01  0.2590E+01
 0.2771E+01  0.2868E+01  0.2831E+01  0.3007E+01  0.2943E+01
 0.2809E+01  0.2614E+01  0.2855E+01  0.2770E+01  0.2874E+01
 0.2988E+01  0.2900E+01  0.2711E+01  0.2566E+01  0.2888E+01
 0.2738E+01  0.2847E+01  0.2897E+01  0.3131E+01  0.2752E+01
         1          40
 0.3000E-03  0.4500E-04
INITIAL CRACK SIZE, AI
 0.2226E-03  0.3291E-03  0.2797E-03  0.2715E-03  0.3743E-03
 0.2921E-03  0.2942E-03  0.2591E-03  0.2747E-03  0.2515E-03
 0.3127E-03  0.3614E-03  0.3482E-03  0.2739E-03  0.3140E-03
 0.2827E-03  0.2768E-03  0.2971E-03  0.3925E-03  0.2652E-03
 0.4206E-03  0.2786E-03  0.3617E-03  0.2727E-03  0.3325E-03
 0.2552E-03  0.2576E-03  0.3440E-03  0.3047E-03  0.3328E-03
 0.2800E-03  0.3017E-03  0.2586E-03  0.2697E-03  0.2558E-03
 0.3213E-03  0.3738E-03  0.2246E-03  0.3674E-03  0.3441E-03
         1          40
 0.2200E-10  0.2200E-11
MATERIAL PROPERTY, C
 0.1807E-10  0.2346E-10  0.2104E-10  0.2063E-10  0.2557E-10
 0.2167E-10  0.2177E-10  0.1999E-10  0.2079E-10  0.1960E-10
 0.2267E-10  0.2498E-10  0.2436E-10  0.2075E-10  0.2274E-10
 0.2120E-10  0.2090E-10  0.2191E-10  0.2640E-10  0.2031E-10
 0.2765E-10  0.2099E-10  0.2499E-10  0.2069E-10  0.2362E-10
 0.1979E-10  0.1992E-10  0.2417E-10  0.2229E-10  0.2364E-10
 0.2106E-10  0.2214E-10  0.1997E-10  0.2054E-10  0.1983E-10
 0.2309E-10  0.2555E-10  0.1817E-10  0.2526E-10  0.2417E-10
         1          40
 0.6200E+03  0.6200E+02
STRESS RANGE, DELSIG
 0.5092E+03  0.6612E+03  0.5931E+03  0.5814E+03  0.7207E+03
 0.6106E+03  0.6135E+03  0.5635E+03  0.5859E+03  0.5524E+03
 0.6390E+03  0.7039E+03  0.6866E+03  0.5848E+03  0.6408E+03
 0.5973E+03  0.5889E+03  0.6175E+03  0.7439E+03  0.5723E+03
 0.7791E+03  0.5915E+03  0.7043E+03  0.5831E+03  0.6658E+03
 0.5578E+03  0.5613E+03  0.6811E+03  0.6281E+03  0.6662E+03
 0.5935E+03  0.6238E+03  0.5627E+03  0.5788E+03  0.5587E+03
 0.6507E+03  0.7200E+03  0.5121E+03  0.7118E+03  0.6812E+03
LOG OF CYCLES TO REACH CRACK SIZE=2.0E-03M,
IVEN STRESS MEAN AMPLITUDE=6.2E+02MPA
 0.4907E+01  0.3774E+01  0.3633E+01  0.3834E+01  0.3334E+01
 0.3867E+01  0.4041E+01  0.3789E+01  0.3705E+01  0.3630E+01
 0.3674E+01  0.3393E+01  0.3253E+01  0.4164E+01  0.3941E+01
 0.3741E+01  0.3698E+01  0.4182E+01  0.3348E+01  0.4144E+01
 0.3228E+01  0.3680E+01  0.3354E+01  0.3514E+01  0.3309E+01
 0.3889E+01  0.4149E+01  0.3390E+01  0.3693E+01  0.3411E+01
 0.3502E+01  0.3518E+01  0.4007E+01  0.4155E+01  0.3775E+01
 0.3666E+01  0.3280E+01  0.3952E+01  0.2868E+01  0.3548E+01
SORTED LOG OF CYCLES
 0.2868E+01  0.3228E+01  0.3253E+01  0.3280E+01  0.3309E+01
 0.3334E+01  0.3348E+01  0.3354E+01  0.3390E+01  0.3393E+01
 0.3411E+01  0.3502E+01  0.3514E+01  0.3518E+01  0.3548E+01
 0.3630E+01  0.3633E+01  0.3666E+01  0.3674E+01  0.3680E+01
 0.3693E+01  0.3698E+01  0.3705E+01  0.3741E+01  0.3774E+01
 0.3775E+01  0.3789E+01  0.3834E+01  0.3867E+01  0.3889E+01
 0.3941E+01  0.3952E+01  0.4007E+01  0.4041E+01  0.4144E+01
 0.4149E+01  0.4155E+01  0.4164E+01  0.4182E+01  0.4907E+01
DESPL PARAMETERS
```

```
21    0    0.5000E+01  0.1000E-03   30
NDS(1),BNDS(2)=   0.2724E+01   0.5153E+01
DF OF LOG OF CURRENT CYCLES,LOG XNF,Y AXIS OF PDF PLOT
 0.0000E+00  0.5049E-01  0.1355E+00  0.2417E+00  0.3572E+00
 0.4706E+00  0.5719E+00  0.6542E+00  0.7124E+00  0.7429E+00
 0.7450E+00  0.7193E+00  0.6684E+00  0.5957E+00  0.5065E+00
 0.4067E+00  0.3027E+00  0.2019E+00  0.1118E+00  0.4101E-01
 0.0000E+00
OUTPUT STATISTICS
-0.2267E+02 -0.3434E+02  0.3913E+01  0.2276E+00
NUMBER OF MISSING VALUES
 0
LOG OF CURRENT CYCLES, LOG XNF
 0.2724E+01  0.5153E+01  0.2846E+01  0.2967E+01  0.3089E+01
 0.3210E+01  0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01
 0.3817E+01  0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01
 0.4424E+01  0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01
 0.5031E+01
ORDERED LOG OF CURRENT CYCLES, LOG,XNF,                       X AXIS PDF, CDF PLOT
 0.2724E+01  0.2846E+01  0.2967E+01  0.3089E+01  0.3210E+01
 0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01  0.3817E+01
 0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01  0.4424E+01
 0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01  0.5031E+01
 0.5153E+01
GCDF PARAMETERS
 2
CDF OF LOG OF CURRENT CYCLES, LOG XNF,                        Y AXIS OF PDF, CDF PLOT
 0.0000E+00  0.3065E-02  0.1435E-01  0.3725E-01  0.7362E-01
 0.1239E+00  0.1872E+00  0.2616E+00  0.3446E+00  0.4329E+00
 0.5232E+00  0.6121E+00  0.6964E+00  0.7731E+00  0.8400E+00
 0.8955E+00  0.9385E+00  0.9692E+00  0.9882E+00  0.9975E+00
 0.1000E+01
ORDERED LOG OF CURRENT CYCLES, LOG XNF,                       X AXIS OF PDF, CDF PLOT
 0.2724E+01  0.2846E+01  0.2967E+01  0.3089E+01  0.3210E+01
 0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01  0.3817E+01
 0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01  0.4424E+01
 0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01  0.5031E+01
 0.5153E+01
 0.2724E+01  0.2846E+01  0.2967E+01  0.3089E+01  0.3210E+01
 0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01  0.3817E+01
 0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01  0.4424E+01
 0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01  0.5031E+01
 0.5153E+01
```

```
11:17:04.4303   0.0000  CSP   University of Texas System Center for High Performance Computing
11:17:04.4306   0.0000  CSP
11:17:04.4308   0.0000  CSP   CRAY X-MP/24  S/N-130/89/26      Austin, Texas            11/29/88
11:17:04.4311   0.0000  CSP
11:17:04.4314   0.0000  CSP   CRAY Operating System            COS 1.16 BF2  Assembled 10/24/88
11:17:04.4317   0.0000  CSP
11:17:04.4319   0.0000  CSP
11:17:04.8613   0.0000  CSP   JOB,JN=RANDM22,US=USAQ530,RT=60,T=30,MFL=3000000.
11:17:04.8856   0.0008  CSP   ACCOUNT,UPW=.
11:17:07.2151   0.1023  CSP   lib,dn=imslib.
11:17:11.0050   0.1357  PDM   PD000 - PDN = IMSLIB            ID = STATIC     ED =     4  OWN = LIBRARY
11:17:11.0055   0.1357  PDM   PD000 - ACCESS  COMPLETE
11:17:11.0311   0.1373  CSP   FETCH,DN=AAA,TEXT=.
11:17:12.3536   0.1374  SCP     VAX TO CRAY: %SYSTEM-S-NORMAL, normal successful completion
11:17:12.3538   0.1374  SCP     VAX TO CRAY: FILE=$1$DUA2:[USAQ530.RANDOM2.NEWRAND2]NORMAL.INP;12
11:17:12.3541   0.1374  SCP     VAX TO CRAY: 248 BYTES TRANSFERRED
11:17:12.9106   0.1374  SCP   SS004 - DATASET RECEIVED FROM FRONT END
11:17:12.9672   0.1374  CSP   ASSIGN,DN=AAA,A=FT05.
11:17:12.9695   0.1376  CSP   ASSIGN,DN=RRR,A=FT34.
11:17:12.9722   0.1378  CSP   ASSIGN,DN=SSS,A=FT35.
11:17:12.9749   0.1382  CSP   ACCESS,DN=$BLD,PDN=NR2BLD,ID=SMBOYCE.
11:17:13.0130   0.1383  PDM   PD000 - PDN = NR2BLD            ID = SMBOYCE    ED =     3  OWN = USAQ530
11:17:13.0132   0.1383  PDM   PD000 - ACCESS  COMPLETE
11:17:13.0375   0.1386  CSP   SEGLDR,CMD='LIB=IMSLIB;DUPENTRY=IGNORE',GO.
11:17:13.1083   0.1393  USER  SG000 - SEGLDR VERSION 3.1  - 05/24/88     (CHG-12/31/86)
11:18:01.3209   7.6090  USER  SG001 - BEGIN EXECUTION
11:18:01.4544   7.6735  USER  UT010 - STOP            in $MAIN
11:18:01.4660   7.6737  CSP   DISPOSE,DN=RRR,SDN=OUT1.
11:18:01.5833   7.6739  CSP   DISPOSE,DN=SSS,SDN=OUT2.
11:18:01.7168   7.6740  CSP   END OF JOB
11:18:02.9591   7.6740  SCP     CRAY TO VAX: %RMS-S-NORMAL, normal successful completion
11:18:02.9594   7.6740  SCP     CRAY TO VAX: FILE=$1$DUA2:[USAQ530.RANDOM2.NEWRAND2]OUT1.CPR;1
11:18:02.9596   7.6740  SCP     CRAY TO VAX: 542 BYTES TRANSFERRED
11:18:02.9819   7.6740  SCP     CRAY TO VAX: %RMS-S-NORMAL, normal successful completion
11:18:02.9822   7.6740  SCP     CRAY TO VAX: FILE=$1$DUA2:[USAQ530.RANDOM2.NEWRAND2]OUT2.CPR;1
11:18:02.9825   7.6740  SCP     CRAY TO VAX: 542 BYTES TRANSFERRED
11:18:03.0361   7.6740  CSP
11:18:03.0363   7.6740  CSP
11:18:03.1065   7.6742  USER      JOB NAME -                            RANDM22
11:18:03.1068   7.6742  USER      USER NUMBER -                         USAQ530
11:18:03.1071   7.6742  USER      JOB SEQUENCE NUMBER -                    4788
11:18:03.1074   7.6742  USER
11:18:03.1077   7.6742  USER      TIME EXECUTING IN CPU -               0000:00:07.6742
11:18:03.1080   7.6742  USER      TIME WAITING TO EXECUTE -             0000:00:41.9199
11:18:03.1083   7.6742  USER      TIME WAITING FOR I/O -                0000:00:08.2839
11:18:03.1086   7.6742  USER      TIME WAITING SEMAPHORE -              0000:00:00.0000
11:18:03.1098   7.6743  USER      TIME WAITING IN INPUT QUEUE -         0000:00:00.0765
11:18:03.1101   7.6743  USER      MEMORY * CPU TIME (MWDS*SEC) -               1.79962
11:18:03.1104   7.6743  USER      MEMORY * I/O WAIT TIME (MWDS*SEC) -          1.81682
11:18:03.1107   7.6743  USER      MEMORY * SEM WAIT TIME (MWDS*SEC) -          0.00000
11:18:03.1110   7.6743  USER      MINIMUM JOB SIZE (WORDS) -              52736
11:18:03.1115   7.6743  USER      MAXIMUM JOB SIZE (WORDS) -             472064
11:18:03.1119   7.6743  USER      MINIMUM FL (WORDS) -                    47104
11:18:03.1122   7.6744  USER      MAXIMUM FL (WORDS) -                   466432
11:18:03.1125   7.6744  USER      MINIMUM JTA (WORDS) -                    4096
11:18:03.1128   7.6744  USER      MAXIMUM JTA (WORDS) -                    5632
11:18:03.1131   7.6744  USER      DISK SECTORS MOVED -                     6194
11:18:03.1134   7.6744  USER      FSS SECTORS MOVED -                         0
11:18:03.1137   7.6744  USER      USER I/O REQUESTS -                       467
11:18:03.1140   7.6744  USER      USER I/O SUSPENSIONS -                   1358
11:18:03.1143   7.6744  USER      OPEN CALLS -                               26
11:18:03.1146   7.6744  USER      CLOSE CALLS -                              28
11:18:03.1149   7.6744  USER      MEMORY RESIDENT DATASETS -                  0
```

```
11:18:03.1152    7.6744  USER   TEMPORARY DATASET SECTORS USED -              5
11:18:03.1155    7.6744  USER   PERMANENT DATASET SECTORS ACCESSED -       4792
11:18:03.1158    7.6745  USER   PERMANENT DATASET SECTORS SAVED -             0
11:18:03.1161    7.6745  USER   SECTORS RECEIVED FROM FRONT END -             1
11:18:03.1164    7.6745  USER   SECTORS QUEUED TO FRONT END -                 2
11:18:03.1167    7.6745  USER
11:18:03.1170    7.6745  USER     7.6742 CPU seconds        =    5.7557 RT seconds ( 82%)
11:18:03.1173    7.6745  USER        467 I/O requests       =    0.1401 RT seconds (  2%)
11:18:03.1177    7.6746  USER     1.7996 CM*CPU mwds*sec    =    0.5399 RT seconds (  8%)
11:18:03.1180    7.6746  USER     1.8168 CM*I/O requests    =    0.5450 RT seconds (  8%)
11:18:03.1183    7.6746  USER                                ---------------------
11:18:03.1186    7.6747  USER                       Total   =    6.9807 RT seconds
11:18:03.1189    7.6747  USER
11:18:03.1192    7.6747  USER    RT cost at bid priority 2  = $     0.97
```

YYYYYYYYYY 111111111111111111111111 Digital Equipment Corporation - VAX/VMS Version V4.7 111111111111111111111111 YYYYYYYYYY

NETNONPRIV

OUT1

CPR;1

File _DBA0:[]OUT1.CPR;1 (363,61,0), last revised on 22-NOV-1988 09:10, is a 2 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

Job OUT1 (201) queued to SYS$SBPRT on 22-NOV-1988 09:10 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100, started on printer _TTF7: on 22-NOV-1988 09:10 from queue TTF7.

YYYYYYYYYY 111111111111111111111111 Digital Equipment Corporation - VAX/VMS Version V4.7 111111111111111111111111 YYYYYYYYYY

```
(E12.4,1X,E12.4)
 0.2724E+01   0.0000E+00
 0.2846E+01   0.5049E-01
 0.2967E+01   0.1355E+00
 0.3089E+01   0.2417E+00
 0.3210E+01   0.3572E+00
 0.3332E+01   0.4706E+00
 0.3453E+01   0.5719E+00
 0.3574E+01   0.6542E+00
 0.3696E+01   0.7124E+00
 0.3817E+01   0.7429E+00
 0.3939E+01   0.7450E+00
 0.4060E+01   0.7193E+00
 0.4182E+01   0.6684E+00
 0.4303E+01   0.5957E+00
 0.4424E+01   0.5065E+00
 0.4546E+01   0.4067E+00
 0.4667E+01   0.3027E+00
 0.4789E+01   0.2019E+00
 0.4910E+01   0.1118E+00
 0.5031E+01   0.4101E-01
 0.5153E+01   0.0000E+00
```

ORIGINAL PAGE IS
OF POOR QUALITY

Digital Equipment Corporation - VAX/VMS Version V4.7

NETNONPRIV

OUT2

CPR;1

File _DBA0:[]OUT2.CPR;1 (383,877,0), last revised on 22-NOV-1988 09:10, is a 2 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

Job OUT2 (202) queued to SYS$SBPRT on 22-NOV-1988 09:10 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100, started on printer _TTF6: on 22-NOV-1988 09:10 from queue TTF6.

Digital Equipment Corporation - VAX/VMS Version V4.7

(E12.4,1X,E12.4)

0.2724E+01	0.0000E+00
0.2846E+01	0.3063E-02
0.2967E+01	0.1433E-01
0.3089E+01	0.3723E-01
0.3210E+01	0.7362E-01
0.3332E+01	0.1239E+00
0.3453E+01	0.1872E+00
0.3574E+01	0.2616E+00
0.3696E+01	0.3446E+00
0.3817E+01	0.4329E+00
0.3939E+01	0.5232E+00
0.4060E+01	0.6121E+00
0.4182E+01	0.6964E+00
0.4303E+01	0.7731E+00
0.4424E+01	0.8400E+00
0.4546E+01	0.8955E+00
0.4667E-01	0.9383E+00
0.4789E-01	0.9692E+00
0.4910E-01	0.9882E+00
0.5031E+01	0.9975E+00
0.5153E+01	0.1000E+01

8.0 APPENDIX C

IMSL SUBROUTINE CALLS FROM RANDOM2

1. RNSET - Initializes a random seed for use in the IMSL random number generators.
2. RNNOR - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.
3. RNLNL - Generates pseudorandom numbers from a lognormal distribution.
4. DESPL - Performs nonparametric probability density function estimation by the penalized likelihood method.
5. GCDF - Evaluates a general continuous cumulative distribution function given ordinates of the density.

9.0 APPENDIX D

SAMPLE SAS/GRAPH (VER. 5.16) PROGRAM FOR RANDOM2

```
data a;
INFILE 'OUT1.CPR' FIRSTOBS=2;input x y;
GOPTIONS DEVICE=HP7470;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
         value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'PROBABILITY DENSITY FUNCTION';
   symbol i=spline v=square;
data B;
INFILE 'OUT2.CPR' FIRSTOBS=2;input x y;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
         value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'CUMULATIVE DISTRIBUTION FUNCTION';
   symbol i=spline v=square;
```