UNCLASSIFIED

20000814051

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

**REPORT DOCUMENTATION PAGE**

READ INSTRUCTIONS BEFORE COMPLETING FORM

| | |
|---|---|
| 1. REPORT NUMBER NSWC TR 83-13 | 2. GOVT ACCESSION NO. AD-A158 111 |
| 3. RECIPIENT'S CATALOG NUMBER | |
| 4. TITLE (and Subtitle) FIVE STATISTICAL PROGRAMS IN BASIC FOR DESKTOP COMPUTERS | 5. TYPE OF REPORT & PERIOD COVERED Final |
| | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s) A. R. DiDonato | 8. CONTRACT OR GRANT NUMBER(s) |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Surface Weapons Center (Code K104) Dahlgren, Virginia 22448 | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NIF |
| 11. CONTROLLING OFFICE NAME AND ADDRESS | 12. REPORT DATE Nov. 1982 |
| | 13. NUMBER OF PAGES 100 |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) | 15. SECURITY CLASS. (of this report) UNCLASSIFIED |
| | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)


Five computer programs especially useful in statistics are described and listings in BASIC are given. The listings are generated using the Hewlett-Packard 85 and 9845 desktop computers. The programs supply the values of: (1) the integral of the bivariate circular normal distribution (ND) over an offset circle; (2) the integral of the bivariate elliptical ND over a circle centered at the origin; (3) the integral of the bivariate elliptical ND over an offset circle; (4) the integral of the bivariate correlated

(Continues)


DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (Continued)

elliptical ND over an arbitrary polygon; (5) the maximum likelihood estimates, obtained from quantal experiments, for the mean and variance of a ND.

## FOREWORD

The work described in this report was done in the Space and Surface Systems Division of the Strategic Systems Department.

The author is indebted to Alfred Morris, Dr. Marlin Thomas, Head of the Mathematical Statistics Staff, Dr. Harold Crutcher, consulting Meteorologist, Peter Shugart of U.S. Army Tradoc Systems for helpful discussions.

The author also wishes to thank Alfred Morris for reviewing and designing improvements to CIRCV, ELLCV, and ELLCV3.

O. F. Braxton

O. F. BRAXTON, Head
Strategic Systems Department

## ABSTRACT

Five computer programs especially useful in statistics are described and listings in BASIC are given. The listings are generated using the Hewlett-Packard 85 and 9845 desktop computers. The programs supply the values of: (1) the integral of the bivariate circular normal distribution (ND) over an offset circle; (2) the integral of the bivariate elliptical ND over a circle centered at the origin; (3) the integral of the bivariate elliptical ND over an offset circle; (4) the integral of the bivariate correlated elliptical ND over an arbitrary polygon; (5) the maximum likelihood estimates, obtained from quantal response experiments, for the mean and variance of a ND.

## CONTENTS





Preceding Pages Blank

# I. INTRODUCTION

With the increasing capability of desktops for scientific computation, the need arises to make available in BASIC several important statistical programs which have been operating in Fortran on large computers. These programs, as described herein, although not particularly lengthy, are sophisticated in their mathematical and logical structure; their design for desktops are in keeping with the high standards of the CDC 6700 mathematics subroutine library maintained at Dahlgren, [14].

We proceed to describe in mathematical terms the statistical functions that are involved. Details beyond those given here and in the next section can be found in the references.

The first and second programs, which appear together in one package, are titled CIRCV and GCEF, respectively.

The purpose of CIRCV is to evaluate P(R, d), where

$$P(R, D) = \exp(-D^2/2) \int_0^R \exp(-r^2/2) I_0(rD)\, r\, dr. \tag{1}$$

Here $R = \bar{R}/\sigma_x$, $D = \sqrt{\bar{h}^2 + \bar{k}^2}/\sigma_x$, where $\bar{R}$ is the radius of a circle C in the xy-plane, offset a distance $\sqrt{\bar{h}^2 + \bar{k}^2}$ from the origin. $I_0(u)$ is the modified Bessel function of the first kind of zeroeth order. Statistically, P represents the probability of a shot falling in C, under a bivariate normal distribution with mean zero and with equal standard deviations $\sigma_x$, $\sigma_y$. [5], [6]. The function P is called the Circular Coverage Function.

The objective of GCEF is to evaluate the probability function:

$$F(K, c) = \frac{1}{c} \int_0^K \exp\left(-\frac{B}{2} r^2\right) I_0\left(\frac{A}{2} r^2\right) r\, dr, \tag{2}$$

where

$$\begin{cases} 0 < c \equiv \sigma_y/\sigma_x < 1, & K \equiv \bar{R}/\sigma_x, \\ A \equiv (1-c^2)/2c^2, & B \equiv (1+c^2)/2c^2. \end{cases} \tag{3}$$

The function F(K, c) is known as the Generalized Circular Error Function, [5], [15].

The third program, called ELLCV, is a generalization of the first two. It supplies the value of P, where

$$P = \frac{1}{2\pi\sigma_x\sigma_y} \int_{\bar{h}-\bar{R}}^{\bar{h}+\bar{R}} \exp\left[-\frac{1}{2}\left(\frac{x}{\sigma_x}\right)^2\right] \int_{\bar{k}-\sqrt{\bar{R}^2-(x-\bar{h})^2}}^{\bar{k}+\sqrt{\bar{R}^2-(x-\bar{h})^2}} \exp\left[-\frac{1}{2}\left(\frac{y}{\sigma_y}\right)^2\right] dy\, dx.^* \tag{4}$$

Statistically, P represents the probability of a shot falling, under a bivariate normal distribution with mean zero and with standard deviations $\sigma_x$ and $\sigma_y$ in the x and y directions, in a circle C, centered at $(\bar{h}, \bar{k})$ with radius $\bar{R}$, i.e.,

$$C:\ (x-\bar{h})^2 + (y-\bar{k})^2 = \bar{R}^2. \tag{5}$$

We call $P(\bar{R}, \bar{h}, \bar{k}, \sigma_x, \sigma_y)$ the Elliptical Coverage Function, [3], [4].

ELLCV is slow in comparison with CIRCV or GCEF. Thus, we also give listings for the program ELLCV3 which supplies $P(\bar{R}, \bar{h}, \bar{k}, \sigma_x, \sigma_y)$ at reduced accuracy but at roughly one-half the computing time per case of ELLCV.

The fourth program is named POLYCV, it makes available the values of $P(\Pi)$ and $A(\Pi)$ or $P(A1)$, where

$$P(\Pi) = \iint_{\Pi \text{ or } A1} Z(x, y)\, dx\, dy, \tag{6}$$

with

$$Z(x, y) = \frac{(1-c^2)^{-1/2}}{2\pi S_x S_y} \exp\left\{-\left[\left(\frac{x-M_x}{S_x}\right)^2 - 2c\left(\frac{x-M_x}{S_x}\right)\left(\frac{y-M_y}{S_y}\right) + \left(\frac{y-M_y}{S_y}\right)^2\right]\Big/ 2(1-c^2)\right\}, \tag{7}$$

and $A(\Pi)$ represents the area of $\Pi$. Here $\Pi$ denotes an arbitrary polygon in the xy-plane which is defined by the coordinates of its vertices $(x_i, y_i)$, $i = 1, 2 \cdots, N$. A1 denotes a semi-infinite angular region in the plane (see Figures 6 and 7, page 17). The integrand of (6), given by (7), represents a correlated bivariate normal density function with mean $(M_x, M_y)$ and covariance matrix

$$\begin{pmatrix} S_x^2 & cS_xS_y \\ cS_xS_y & S_y^2 \end{pmatrix}, \tag{8}$$

with correlation coefficient c, $(|c| < 1)$. Details of the analysis for POLYCV are given in [9], [10], [11], [12].

The last program is named MLEQRE, which stands for "Maximum Likelihood Estimates from Quantal Response Experiments." An estimated mean $\mu$ and an estimated standard deviation $\sigma$ of a

*Note that (1) and (2) follow from (4) by transforming (x, y) to polar coordinates $(r, \theta)$ and setting $\sigma_x = \sigma_y$ for (1), and $\bar{h} = \bar{k} = 0$ for (2).

normal distribution obtained from quantal responses are supplied by MLEQRE. It also gives the associated covariance matrix elements and a plot of the 50 and 95% confidence ellipses. The estimates $\mu$, $\sigma$ are taken as those unique values of the independent variables u, s, respectively, which maximize the likelihood function

$$F \equiv \prod_{i=1}^{\bar{N}} P(s_i) \prod_{j=1}^{\bar{M}} Q(t_j), \tag{9}$$

where

$$Z(v) \equiv \frac{1}{\sqrt{2\pi}} \exp(-v^2/2), \quad P(v) = \int_{-\infty}^{v} Z(r)\,dr, \quad Q(v) = \int_{v}^{\infty} Z(r)\,dr, \tag{10}$$

$$s_i = (a_i - u)/s, \quad t_j = (b_j - u)/s. \tag{11}$$

The $a_i$, $b_j$ are input stimuli from a set of quantal experiments. They are called "successes" and "failures," respectively, [7], [8], where $\bar{N}$ denotes the number of successes and $\bar{M}$ the number of failures.

In the next section we discuss each program and how to use it. The third section contains the program listings in BASIC with sample outputs.

## II. MATHEMATICAL DESCRIPTION OF PROGRAMS

In this section, the five programs, introduced in the previous section, namely

CIRCV
GCEF
ELLCV (ELLCV3)
POLYCV
MLEQRE

are described in mathematical terms.

All of the programs, except POLYCV, contain a subroutine for computing the complementary error function, erfc (·), to a preset relative accuracy using Cody's rational functions, [2], i.e.,

$$\operatorname{erfc}(x) = 1 - x \sum_{i=0}^{J-1} P9(J-i)x^{2i} \Big/ \sum_{i=0}^{J-1} Q9(J-1-i)x^{2i}, \quad Q9(0) \equiv 1, \quad |x| \leq 1/2, \tag{12}$$

$$\operatorname{erfc}(x) = e^{-x^2} \sum_{i=0}^{J1-1} R9(J1-i)x^{i} \Big/ \sum_{i=0}^{J1-1} S9(J1-1-i)x^{i}, \quad S9(0) \equiv 1, \quad 1/2 < x \leq 4, \tag{13}$$

$$\operatorname{erfc}(x) = \frac{e^{-x^2}}{x}\left[\frac{1}{\sqrt{\pi}} + \frac{1}{x^2}\sum_{i=0}^{J2-1} V9(J2-i)x^{-2i} \bigg/ \sum_{i=0}^{J2-1} W9(J2-1-i)x^{-2i}\right],\ W9(0) \equiv 1,\ x > 4. \tag{14}$$

We also use the fact that

$$\operatorname{erfc}(x) = 2 - \operatorname{erfc}(-x). \tag{15}$$

The Cody coefficients are stored in arrays for each of the five programs as noted in (12)-(14). For example in MLEQRE they are listed, starting with P9(1), in lines 225-260 of the HP-85 listing (see page 91). These sets of coefficients may vary from one program to another depending on the accuracy desired.

The subroutine CIRCV provides the value of P, where

$$P = \frac{1}{2\pi}\iint_C \exp\left[-\frac{1}{2}(x^2+y^2)\right] dx\,dy, \tag{16}$$

and C denotes the circle:

$$C: (x-\bar{h})^2 + (y-\bar{k})^2 = \bar{R}^2, \tag{17}$$

i.e., as noted earlier, C is the circle in the xy-plane with center at $(\bar{h}, \bar{k})$ and radius $\bar{R}$. The normalized offset distance from $(\bar{h}, \bar{k})$ to the origin is denoted by

$$D = \sqrt{\bar{h}^2 + \bar{k}^2}/\sigma_x. \tag{18}$$

The integration of (16) is carried out in polar coordinates as indicated by (1). The derivation of (1) from (16) is given in [5].

Two sets of recurrence relations are used; the choice depending on the value of RD, $(R \equiv \bar{R}/\sigma_x)$.

For

$$\underline{0 \leq RD \leq 7.0},$$

we have

$$P = \sum_{n=0}^{\infty} g_n k_n \tag{19}$$

$$g_n \equiv \frac{1}{n!}\left(\frac{R^2}{2}\right)^{n+1}\int_0^1 u^n e^{-uR^2/2}\,du, \quad k_n \equiv \left(\frac{D^2}{2}\right)^n \frac{1}{n!} e^{-D^2/2} \tag{20}$$

$$\begin{cases} g_n = g_{n-1} - \dfrac{1}{n!}\left(\dfrac{R^2}{2}\right)^n e^{-R^2/2}, & g_0 = (1 - e^{-R^2/2}) \\[2ex] k_n = \left(\dfrac{D^2}{2}\right)\dfrac{1}{n}\, k_{n-1}, & k_0 = e^{-D^2/2} \end{cases} \tag{21}$$

Then (20) and (21) can be used to obtain

$$\begin{cases} n = n + 1 \\[1ex] K_n \equiv \dfrac{1}{n!}\left(\dfrac{R^2}{2}\right)^n e^{-R^2/2}\, k_n = \dfrac{1}{n^2}\left(\dfrac{RD}{2}\right)^2 K_{n-1}, \\[1ex] T_n \equiv g_n k_n = \left(\dfrac{D^2}{2}\right)\dfrac{1}{n}\, T_{n-1} - K_n \\[1ex] S1 = S1 + T_n, \quad S2 = S2 + K_n \\[1ex] K_0 = \exp\left[-(R^2 + D^2)/2\right], \quad T_0 = \begin{cases} \dfrac{R^2}{2}\exp(-D^2/2) & \text{if } \dfrac{R^2}{2} < 5 \times 10^{-4} \\[2ex] \exp(-D^2/2) - K_0 & \text{if } \dfrac{R^2}{2} > 5 \times 10^{-4}. \end{cases} \end{cases} \tag{22}$$

The recurrence relations in (22) are cycled starting with $n = 0$, $S1 = T_0$, $S2 = K_0$. At the end of each cycle, test

(a) $n > (RD/\sqrt{2}) - 1$.

Cycle (22) until (a) is true, then test

(b) $T_n < \epsilon$.

Continue to cycle (22) and test only (b) at the end of each cycle until (b) holds. Then, with 6-decimal-digit accuracy.

$$\begin{cases} P = S1 \\[1ex] \dfrac{\partial P}{\partial R} = R * S2. \end{cases} \tag{23}$$

The function $\partial P/\partial R$ is available at virtually no additional computation. If R is desired for a fixed P and D, then $\partial P/\partial R$ can be used to find R by the Newton-Raphson procedure.

The recurrence relations given above, based on [1], yield a slightly faster algorithm than those given by (22) and (23) in [5]. The resulting algorithm and test (a), as described above, are slightly improved over those in [1]*.

*We wish to thank Peter Shugart for bringing [1] to our attention.

For large RD the above algorithm is inefficient. Consequently, the following algorithm, as developed in [5], [6], is used if

$$\underline{RD > 7.0.}$$

Initially, we compute

$$\begin{cases} X_1 = \frac{1}{2} \frac{1}{\sqrt{2RD}} \frac{2}{\sqrt{\pi}} \exp\left[-(R-D)^2/2\right] \\ M_1 = \frac{(R+D)}{\sqrt{2}} \frac{1}{\sqrt{2RD}} \operatorname{erfc}\left(\frac{|R-D|}{\sqrt{2}}\right) \\ S1 = M_1, \quad S2 = X_1. \end{cases} \tag{24}$$

Then starting with n = 0:

$$\begin{cases} n = n + 1 \\ X_{2n+1} = \left(\frac{2n-1}{2n}\right) \frac{1}{4RD} X_{2n-1}{}^{*} \\ M_{2n+1} = |R^2 - D^2| X_{2n+1} - \frac{(R-D)^2}{4RD} \left(\frac{2n-1}{2n}\right) M_{2n-1} \\ X_{2n+1} = (2n-1) X_{2n+1} \\ S1 = S1 + M_{2n+1}, \quad S2 = S2 + X_{2n+1}. \end{cases} \tag{25}$$

Iterate (25) until

$$M_{2n+1} \leqslant \epsilon. \tag{26}$$

When (26) is satisfied, continue to cycle

$$\begin{cases} n = n + 1 \\ X_{2n+1} = \frac{(2n-1)^2}{2n} \frac{1}{4RD} X_{2n-1} \\ S2 = S2 + X_{2n+1} \end{cases} \tag{27}$$

until

$$X_{2n+1} \leqslant \epsilon. \tag{28}$$

*The quantity $X_{2n+1}$ as given above differs by a factor of 1/4RD from $\bar{X}_{2n+1}$ as given in (31), (32), (34) of [6].

When (28) holds, then with 6 decimal-digit accuracy

$$\begin{cases} P = \frac{1}{2} \,|1 + \operatorname{sgn}(R - D) - S2 - \operatorname{sgn}(R - D) * S1| \\ \frac{\partial P}{\partial R} = R * S2 , \end{cases} \tag{29}$$

where $\operatorname{sgn}(R - D) \equiv 0$ if $R = D$.

We note that Cody's algorithm for erfc $(\cdot)$ is only needed for arguments $\leqslant 4$. There is no need to consider larger arguments because if $|R - D|/\sqrt{2} > 4$ then $P = 1$ or $0$, within 6 decimal digits, depending on the sign of $(R - D)$, (see lines 205, 210 of the HP-85 listing). In case it is desired to achieve greater accuracy, then for arguments larger than 4, use in place of $M_1$ in (24)

$$M_1 = \left(\frac{R + D}{\sqrt{2}}\right) \frac{\sqrt{2}}{|R - D|} X_1 .$$

Of course, other changes would also be needed, such as changing the value of $\epsilon$ and the right hand side of $RD > 7$.

The subroutine GCEF is included in the same program package with CIRCV, because the Generalized Circular Error Function $F(K, c)$ can be obtained using $P(R, D)$. Indeed, we have from (16) and (17) in [5]

$$\begin{cases} P(R, D) - P(D, R) = \operatorname{sgn}(R - D) F(K, c) \\ P(R, D) + P(D, R) = 1 - \exp\,[-(R^2 + D^2)/2]\, I_0(RD) . \end{cases} \tag{30}$$

But from (33) below $R - D > 0$, hence

$$F(K, c) = |2P(R, D) - 1 + \exp\,[-(R^2 + D^2)/2]\, I_0(RD)| . \tag{31}$$

Note that the product in (31) of the exponential and Bessel function is given by S2 in the computation of $P(R, D)$.

The arguments of $F(K, c)$ are given by

$$\begin{cases} K = \bar{R}/\sigma_x , \quad (\bar{R} \text{ comes from (3)}) , \\ 0 < c = \sigma_y/\sigma_x \leqslant 1 , \end{cases} \tag{32}$$

where R and D in (31) are expressed in terms of K and c by

$$R = K\left(\frac{1 + c}{2c}\right), \quad D = K\left(\frac{1 - c}{2c}\right), \quad c \neq 0 , \tag{33}$$

and

$$RD = K^2\left(\frac{1-c^2}{2c^2}\right)\Big/2, \qquad \frac{R^2+D^2}{2} = K^2\left(\frac{1+c^2}{2c^2}\right)\Big/2. \tag{34}$$

It can also be shown that

$$\frac{\partial F}{\partial K} = (K/c)S2, \qquad c \neq 0. \tag{35}$$

The case c = 0 is treated separately in GCEF with

$$\begin{cases} F(K,0) = 1 - \mathrm{erfc}\,(K/\sqrt{2}) \\ \dfrac{\partial F}{\partial K}(K,0) = \sqrt{\dfrac{2}{\pi}}\,S2, \quad \left(= \sqrt{\dfrac{2}{\pi}}\,e^{-K^2/2}\right). \end{cases} \tag{36}$$

We note if $c > 1$, then simply redefine K and c so that

$$\begin{cases} K = \bar{R}/\sigma_y, \quad (\bar{R} \text{ from } (3)), \\ c = \sigma_x/\sigma_y, \end{cases} \tag{37}$$

i.e., interchange $\sigma_x$ and $\sigma_y$, (see (2)).

The program input variables are R, D, V. The output variables are P, S2.

**CIRCV**

*Input:* $\bar{R}/\sigma_x$, $D/\sigma_x$, $1(=V)$, $(\sigma_x > 0)$.

*Output:* If $\bar{R}/\sigma_x$ and $D/\sigma_x \geq 0$, then $P = P(R, D)$, $S2 = 1/R\ \partial P/\partial R$. If R and/or D $< 0$, then $P = -1$ indicating unacceptable input.

**GCEF**

*Input:* $\bar{R}/\sigma_x(=K)$, $\sigma_y/\sigma_x(=c)$, $0(=V)$, $(\sigma_x, \sigma_y > 0)$.

*Output:* If $\bar{R}/\sigma_x \geq 0$ and $0 \leq \sigma_y/\sigma_x \leq 1$, then $P = F(K, c)$ and $S2 = c/K\ \partial F/\partial K (c \neq 0)$; $S2 = \sqrt{\pi/2}\ \partial F/\partial K$ if $c = 0$. If $\bar{R}/\sigma_x < 0$ or $|\sigma_y/\sigma_x - 1/2| > 1/2$ then $P = -1$ indicating unacceptable input. If $\sigma_y > \sigma_x$ the user must interchange $\sigma_x$ and $\sigma_y$.

The subroutine ELLCV provides the value of

$$P = \frac{1}{2\pi\sigma_x\sigma_y}\iint_C \exp\left\{-\frac{1}{2}\left[\left(\frac{x}{\sigma_x}\right)^2 + \left(\frac{y}{\sigma_y}\right)^2\right]\right\} dx\,dy\,; \tag{38}$$

C denotes the circle given by (17). It is shown in [3] that (4) can be expressed in the form

$$P = \frac{R}{\sqrt{2\pi}} \int_0^1 \left[\exp(-X_0^2/2) + \exp(-X_1^2/2)\right]\left[\operatorname{erfc}(y_0) - \operatorname{erfc}(y_1)\right] t\,dt, \tag{39}$$

where

$$\left\{\begin{array}{ll} X_0 \equiv h - R(1-t^2), & y_0 \equiv \dfrac{k - \tilde{R} t \sqrt{2 - t^2}}{\sqrt{2}}, \\[2ex] X_1 \equiv h + R(1-t^2), & y_1 \equiv \dfrac{k + \tilde{R} t \sqrt{2 - t^2}}{\sqrt{2}}, \\[2ex] R \equiv \bar{R}/\sigma_x, \quad \tilde{R} \equiv \bar{R}/\sigma_y, & h \equiv \bar{h}/\sigma_x, \quad k \equiv \bar{k}/\sigma_y. \end{array}\right. \tag{40}$$

Without loss of generality $\bar{h}$ and $\bar{k}$ are assumed to be non-negative.

The average computation time to evaluate P from (39) is an order of magnitude larger than the average time for CIRCV or GCEF. Hence, a number of tests are used to determine if $P < \epsilon$ or $P > 1 - \epsilon$ in which case P is set to zero or one, respectively. Let $H^2 \equiv \bar{h}^2 + \bar{k}^2$, $\sigma = \max(\sigma_x, \sigma_y)$.

*Test #1:* If $\bar{R}^2 < 2\epsilon\sigma_x\sigma_y$ then $P < \epsilon$.

*Test #2:* If $\bar{R} - \bar{h} + A\sigma_x < 0$ or $\bar{R} - \bar{k} + A\sigma_y < 0$ then $P < \epsilon$.

*Test #3:* If $\bar{R} > \sqrt{\bar{h}^2 + \bar{k}^2} + A1 * \sigma$, then $P > 1 - \epsilon$.

*Test #4:* If $H^2 > \bar{R}^2$ and if

$$\bar{R}^2 \exp\left\{-\frac{1}{2}\left[(H - \bar{R})/\sigma\right]^2\right\} < 2\epsilon\sigma_x\sigma_y, \quad \text{then} \quad P < \epsilon.$$

The value of A1 is chosen so that E (see Figure 1) contains $1 - \epsilon$ of the distribution; A is chosen in a similar way with E replaced by a rectangle centered at the origin with sides of length $2A\sigma_x$ and $2A\sigma_y$ along the x and y axes, respectively, (See (50) and (51)).

Test #1 follows by taking $H = 0$ and considering small R. Tests #2 and #3 are covered in [3]. Test #4 follows by using the fact that

$$\exp\left\{-\frac{1}{2}\left[\left(\frac{x}{\sigma_x}\right)^2 + \left(\frac{y}{\sigma_y}\right)^2\right]\right\} < \exp\left[-\frac{1}{2\sigma^2}(x^2 + y^2)\right].$$

Tests #5 and #6, given below, are more subtle. We discuss test #5 first.

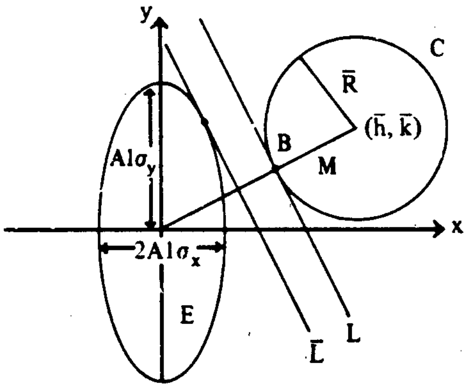



Figure 1. Test #5

Consider Figure (1). A tangent line L is drawn at point B where the ray M, from the origin to $(\bar{h}, \bar{k})$, intersects C; consider also a line $\bar{L}$ with the same slope as L tangent to E, the ellipse with center at the origin with major and minor axes $2A1\sigma_y$, $2A1\sigma_x$. The distance from the origin to B is given by

$$d(L) = H - \bar{R} > 0 \tag{41}$$

and the normal distance from $\bar{L}$ to the origin is given by

$$d(\bar{L}) = A1\sigma_x[(\bar{h}^2 + \alpha^2\bar{k}^2)/H^2]^{1/2}, \quad \alpha \equiv \sigma_y/\sigma_x. \tag{42}$$

Hence, if

$$d(L) > d(\bar{L}) \tag{43}$$

then $P < \epsilon$. This test can be formulated without using square roots and is called Test #5.

*Test #5:* If

$$y \equiv H^2 - \bar{R}^2 - A1^2\sigma_x^2\left(\frac{\bar{h}^2 + \alpha^2\bar{k}^2}{H^2}\right) > 0 \tag{44}$$

and

$$y^2 > 4\bar{R}^2A1^2\sigma_x^2\left(\frac{\bar{h}^2 + \alpha^2\bar{k}^2}{H^2}\right) \tag{45}$$

then $P < \epsilon$.

For test #6 consider Figure (2). Here $\bar{L}$ denotes the tangent line to E at D where ray M intersects E at D; L denotes the line tangent to C which intersects M and which is parallel to $\bar{L}$. In this case

$$T \equiv d(L) - d(\bar{L}) = H\{1 - \tilde{R}\sqrt{F/D} - A1/\sqrt{D}\} \tag{46}$$

where

$$F \equiv \alpha^2h^2 + k^2, \quad D \equiv h^2 + k^2, \quad \tilde{R} \equiv \bar{R}/\sigma_y.$$

If $T > 0$, then $P < \epsilon$. Without using square roots, we have:

*Test #6:* If

$$y \equiv D - \tilde{R}^2F/D - A1^2 > 0, \tag{47}$$

and if

$$y^2 \geqslant 4A1^2\tilde{R}^2F/D, \tag{48}$$

then $P \leqslant \epsilon$.

If none of the above tests are applicable, then a Gaussian numerical integration is used to evaluate P from (39). In this case it is advantageous, if possible, to reduce the interval of integration. By methods similar to those described on pp 7-9 of [3], an interval of integration $[e_0, e_1] \subseteq [0, 1]$ is determined such that

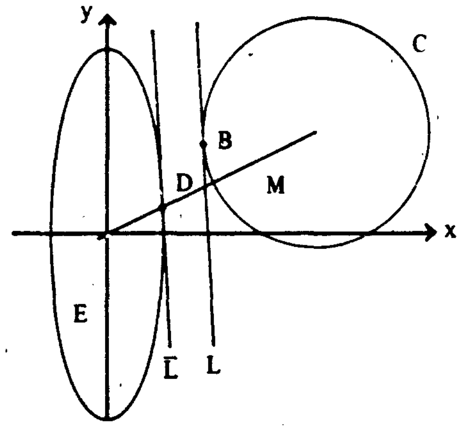



**Figure 2. Test #6**

$$[e_0, e_1] = \min\left\{[\bar{e}_0, \bar{e}_1], [\bar{\bar{e}}_0, \bar{\bar{e}}_1]\right\}, \tag{49}$$

where

$$\bar{e}_0 \equiv \begin{cases} \sqrt{\dfrac{R-h-A3}{R}} & \text{if} \quad h+A3<R, \\ 0 & \text{if} \quad h+A3 \geqslant R. \end{cases} \qquad \bar{\bar{e}}_0 \equiv \begin{cases} \sqrt{\dfrac{\tilde{R}-k-A3}{\tilde{R}}} & \text{if} \quad k+A3<\tilde{R}, \\ 0 & \text{if} \quad k+A3 \geqslant \tilde{R}. \end{cases} \tag{50}$$

$$\bar{e}_1 \equiv \begin{cases} \sqrt{\dfrac{R-h+A3}{R}} & \text{if} \quad 0<h-A3<R, \\ 1 & \text{if} \quad h \leqslant A3, \\ 0 & \text{if} \quad h-A3 \geqslant R. \end{cases} \qquad \bar{\bar{e}}_1 \equiv \begin{cases} \sqrt{\dfrac{\tilde{R}-k+A3}{\tilde{R}}} & \text{if} \quad 0<k-A3<\tilde{R}, \\ 1 & \text{if} \quad k \leqslant A3. \\ 0 & \text{if} \quad k-A3 \geqslant \tilde{R}. \end{cases} \tag{51}$$

If $[\bar{e}_1 - \bar{e}_0] > [\bar{\bar{e}}_1 - \bar{\bar{e}}_0]$, then $\sigma_x$ is interchanged with $\sigma_y$ and h with k. This is equivalent to reversing the order of integration in (39).

In order to retain efficiency in the Gaussian quadrature evaluation of (39), an empirical function N is used to specify *apriori*, the order of the Gaussian process, O(G), to use. It is given by

$$N = \frac{(e_1 - e_0)}{2} \tilde{R} \left[\frac{0.34}{\sigma_x} + \frac{1}{.025|\tilde{R} - \tilde{k}| + 5\sigma_y}\right]. \tag{52}$$

Then

$$\begin{aligned} N \geqslant 2.75 &\rightarrow O(G) = 24; \\ 1.35 \leqslant N < 2.75 &\rightarrow O(G) = 20; \\ 0.75 \leqslant N < 1.35 &\rightarrow O(G) = 16; \\ 0.35 \leqslant N < 0.75 &\rightarrow O(G) = 12; \\ 0.15 \leqslant N < 0.35 &\rightarrow O(G) = 8; \\ 0 < N < 0.15 &\rightarrow O(G) = 6. \end{aligned} \tag{53}$$

For example if, from (52), N = 1.31, then O(G) = 16. This means 16 Gaussian abscissae and weights are used to evaluate the right hand side of (39). The abscissae, starting with O(G) = 6 are stored in array X(i), and the corresponding weights are stored in array Y(i). Since the abscissae and weights have certain symmetry properties about zero only half of them are actually stored. Thus the right hand side of (39) is approximated to within $2\epsilon$ by

$$P \cong \frac{1}{\sqrt{2\pi}}\left(\frac{e_1 - e_0}{2}\right) R \sum_{i=-M/2}^{M/2} w_i f_i p_i t_i, \quad M = O(G), \quad w_0 \equiv 0, \tag{54}$$

where

$$\begin{cases} w_i \equiv i\text{th Gaussian weight}, \quad w_i = w_{-i} \\ t_i = \dfrac{(e_1 - e_0)}{2}(1 + x_i) + e_0, \quad x_i \equiv i\text{th Gaussian abscissa}, \quad x_i = -x_{-i} \end{cases} \tag{55}$$

$$\begin{cases} p_i \equiv \{\mathrm{erfc}\,[y_0(t_i)] - \mathrm{erfc}\,[y_1(t_i)]\}, \\ f_i = \left\{\exp\left[-X_0^2(t_i)/2\right] + \exp\left[-X_1^2(t_i)/2\right]\right\}, \quad (\text{See } (40)). \end{cases} \tag{56}$$

Further reductions in computing time can often be realized by the following:

(a) In the determination of $[e_1 - e_0]$, it can be shown that if $h > A$ and if $R > h - A$ or $R > h + A$ then the second exponential in (39) is negligible and can be dropped. Thus a variable H5 is introduced and set to one if the exponential has been dropped.

(b) An exponential in (39) can also be dropped if the absolute value of its argument exceeds

$$Z8 \equiv \log\left[\frac{1}{\sqrt{\pi}}\left(\frac{e_1 - e_0}{2}\right)\frac{R}{\sqrt{2}}\right] + \begin{cases} 14.51 \text{ (for ELLCV)} \\ 7.70 \text{ (for ELLCV3)}. \end{cases} \tag{57}$$

This feature is not included in ELLCV or ELLCV3 at present.

(c) In case $\bar{k} = 0$, the quantity $p_i$ in (56) is replaced by

$$p_i = 2\{1 - \mathrm{erfc}\,[y_1(t_i)]\}. \tag{58}$$

Thus, only one instead of two erfc functions are required for each i. (If $\sigma_x = \sigma_y$, then by circular symmetry, $\bar{h}$ can be replaced by $\sqrt{\bar{h}^2 + \bar{k}^2}$ and $\bar{k}$ by zero.)

(d) The argument $y_0(y_1)$ of erfc $(\cdot)$ is a decreasing (increasing) function of t. Hence if $\leqslant -A2\,(y_1(\bar{t}) \geqslant A2)$ then $p_i \cong 2(1 - \epsilon_1)$ (erfc $[y_1(\bar{t})] \cong \epsilon_1$) for all $t > \bar{t}$. Two meters are introduced to take advantage of this situation when it occurs. Variable Z is set to one for the smallest i (say =j) on $[-M/2, M/2]$ for which $y_0(t_j) \leqslant -A2$ so that $p_i$ is replaced by $2(1 - \epsilon_1)$ for all $t_i \geqslant t_j$. Similarly a variable Z3 is set to one for the smallest i (say =j) on $[-M/2, M/2]$ for which $y_1(t_j) \geqslant A2$ so that erfc $[y_1(t_i)]$ is replaced by $\epsilon_1$ for all $t_i \geqslant t_j$. Note: $t_i < t_{i+1}$ by the way the Gaussian abscissae are ordered.

In addition to ELLCV which gives P to at least 6-decimal-digit accuracy, a listing is also included for the program ELLCV3 which gives P to at least 3-decimal-digits. The latter differs from ELLCV in the assignment of O(G) by (53), by using fewer Cody coefficients to compute erfc (·) to less accuracy, in the values of the constants A, $A_1$, $A_2$, $A_3$, $\epsilon_1$.

The values of these parameters are given in the table below.

| | ELLCV, $\epsilon = 5(-7)$ | ELLCV3, $\epsilon = 5(-4)$ |
|---|---|---|
| A | 4.892 | 3.291 |
| A1 | 5.387 | 3.89895 |
| A2 | 3.8775 | 2.898 |
| A3 | 5.16 | 3.70 |
| $\epsilon_1$ | 1.04(−8) | 1.5(−5) |

The values of O(G) for ELLCV3 are determined by N from (52) and the following inequalities:

$$\begin{cases} N > 2 & \Rightarrow O(G) = 8 \\ 0.675 \leq N < 2 & \Rightarrow O(G) = 6 \\ 0.5 \leq N < 0.675 & \Rightarrow O(G) = 4 \\ N < 0.5 & \Rightarrow O(G) = 3. \end{cases} \tag{59}$$

The inputs to these subroutines are: $\bar{R}, \bar{h}, \bar{k}, \sigma_x, \sigma_y$. The output is designated by P. The value of $\bar{R}$ must be non-negative; $\sigma_x$ and $\sigma_y$ must be positive. Tests are not included for these requirements; no error parameter is used.

Accuracy: P is given approximately to within $2\epsilon = 10^{-6}$ (or $10^{-3}$ for ELLCV3).

Note: Constraints imposed on h, k, $\sigma_x/\sigma_y$ in [3] are no longer necessary.

The subroutine POLYCV supplies the value of P(Π) as expressed by the double integral in (6). The evaluation of (6) is simplified by using the transformation

$$\bar{x} = \left[\frac{x - M_x}{S_x} - c\,\frac{y - M_y}{S_y}\right] \Big/ \sqrt{1 - c^2}\,, \quad \bar{y} = \frac{y - M_y}{S_y}\,. \tag{60}$$

Then

$$P(\Pi) = \frac{1}{2\pi} \iint_{\bar{\Pi}} \exp\left[-\frac{1}{2}\,(\bar{x}^2 + \bar{y}^2)\right] d\bar{x}\, d\bar{y} \tag{61}$$

where Π of (6) is transformed to $\bar{\Pi}$ of (61) by (60). We call $\bar{\Pi}$ the "transformed polygon." The program determines the vertices $(\bar{x}_i, \bar{y}_i)$ of $\bar{\Pi}$ from (60) and evaluates P(Π) from (61). POLYCV can also be used to evaluate the double integral in (6) over a semi-infinite angular region. This will be discussed further below.

It is important in order to use POLYCV to understand how $\Pi$ must be specified. We say $\Pi$ is *positively oriented* (PO) if it is a simple polygon or the limit of a sequence of simple polygons as defined on page 9 of [11], and if its vertices are ordered so that the area of $\Pi$ is on one's left as the segments of $\Pi$ are traversed continuously in the order the vertices are given. If the area is on the right, $\Pi$ is said to be negatively oriented (NO) and P will be negative. In case $\Pi$ has vertices which occur more than once both (PO) and (NO) regions can occur. Self-intersecting (SI) polygons, as described on pages 13-17 [11], can also be handled. However P and/or the area can be negative. The interpretation of such results is left to the user.

Two examples, to help clarify these ideas, are given in Figures 3 and 4 below. In Figure 3 we have an example of a simple polygon which is PO. The probability P is found over the cross-hatched region. In Figure 4, a polygon is shown with PO and NO regions. The probability is found again over the cross-hatched areas.

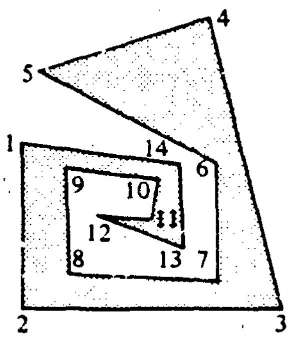



**Figure 3. A Simple (PO) Polygon**

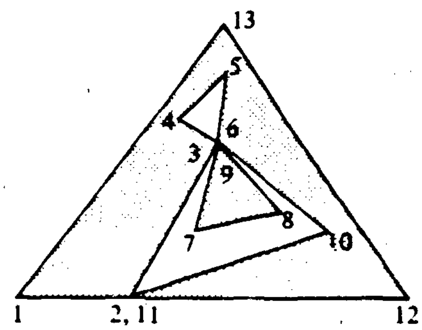



**Figure 4. A Polygon with (PO) and (NO) Regions**

The description given above for specifying $\Pi$ is adequate for most applications. When prescribing a completely arbitrary polygon, all the vertices, points where two segments cross, and initial and terminal points of overlapping segments must be numbered. In certain situations some of these points may not be necessary as shown by the example on page 24 of [11]. However, when in doubt, $\Pi$ should be numbered as just described. More details and examples are given on page 14 of [11].

It is shown in [11], [12] that P over $\Pi$ is given by

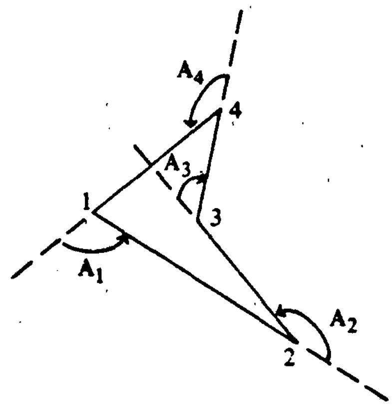
$$P(\Pi) = W - \sum_{i=1}^{N} P(A_i), \tag{62}$$

where N is the number of points specifying $\Pi$, and

$$\begin{cases} W = \Omega/2\pi, \quad (\text{See (75) for } \Omega), \\ P(A_i) \equiv \dfrac{1}{2\pi} \displaystyle\iint_{A_i} \exp\left[-\frac{1}{2}(\bar{x}^2 + \bar{y}^2)\right] d\bar{x}\, d\bar{y}. \end{cases} \tag{63}$$

Here $A_i$ denotes a semi-infinite exterior angular region of $\bar{\Pi}$. In Figure 5, the four exterior angular regions are shown for a simple polygon with N = 4. For i = 1, ..., N, $A_i$ is formed by extending the side $\overline{(i-1, i)}$ from (i) to $\infty$ in the direction from (i − 1) to (i). Note (0) ≡ (N). Similarly side $\overline{(i, i+1)}$ is extended from (i + 1) to $\infty$ in the direction (i) to (i + 1), where (N + 1) ≡ (1). The angle of $A_i$, with its vertex at (i), is measured as positive in the counterclockwise direction and as negative in the clockwise direction.

**Figure 5. Shows Angular Region**




In Figure 5, angular regions $A_1$, $A_2$, $A_4$ have positive measures with $P(A_1)$, $P(A_2)$, $P(A_4)$ positive, angular region $A_3$ has negative measure with $P(A_3) < 0$.

The sum of the angular measures of the $A_i$, i = 1, 2, ..., N appears in (63) and is denoted by $\Omega$.

In order to evaluate $P(A_i)$, advantage is taken of the circular symmetry of the integrand in (63). A semi-infinite straight line L is introduced extending from the origin to $v_i$, the vertex of $A_i$, to $\infty$. Then transforming the integration variables in (63) to polar coordinates $(r, \theta)$ at $v_i$, with $\theta > 0$ when measured counterclockwise from L about $v_i$, we have as derived on pp 2-3 of [9],

$$P(A_i) = \frac{1}{2\pi} \int_{\theta_1}^{\theta_2} \int_0^{\infty} \exp\left[-\frac{1}{2}(R^2 + 2rR\cos\theta + r^2)\right] r\,dr\,d\theta, \tag{64}$$

where R is the distance from the origin to $v_i$, and $\theta_2 - \theta_1 = \Delta\theta$ is the angular measure of $A_i$. Note if $v_i$ is at (0, 0), so that R = 0, then $P(A_i) = \Delta\theta/2\pi$.

Using the fact that

$$\int_0^{\infty} e^{-r^2/2} e^{-rR\cos\theta}\, r\,dr = 1 - 2u\,\mathrm{erfc}(u)/z(u), \tag{65}$$

where

$$u \equiv \frac{R}{\sqrt{2}} \cos\theta, \quad z(u) \equiv \frac{2}{\sqrt{\pi}} \exp(-u^2), \quad \operatorname{erfc}(u) \equiv \int_u^\infty z(t)\,dt, \tag{66}$$

we have

$$P(A_i) = e^{-R^2/2} \left[ \frac{\Delta\theta}{2\pi} - \frac{1}{\pi} \int_{\theta_1}^{\theta_2} u\,[\operatorname{erfc}(u)/z(u)]\,d\theta \right], \quad -\pi < \Delta\theta \leqslant \pi. \tag{67}$$

The evaluation of the right hand side of (67) is achieved by using a minimax polynomial approximation on $[0, C(\delta)]$, i.e., given a $\delta > 0$, a set of real numbers $\{U1_k\}$ are found for a least positive integer K such that

$$\left| \operatorname{erfc}(u) - z(u) \sum_{k=0}^{K-1} U1_{K-k} u^k \right| \leqslant \frac{2}{\sqrt{\pi}} \delta, \quad 0 \leqslant u \leqslant C(\delta). \tag{68}$$

Given a value of $\delta > 0$, the constant $C(\delta)$ is determined so that the value of P(A) in (63) is negligible when A is the angular region with $C(\delta) = R$, with vertex on the positive x-axis and with $\theta_2 = -\theta_1 = \pi/2$. For $\delta = 5 \times 10^{-10}$, $C(\delta) = 6.2$, and $K = 15$.

The coefficients $\{U1_k\}$ for $\delta = 5 \times 10^{-10}$ are given in the HP 9845 POLYCV listing in lines 205–275. Coefficients for some other $\delta$'s are given on page E-4 of [11].

With (68) it is not difficult to show that, within $\delta/\sqrt{\pi}$,

$$P(A_i) = \frac{e^{-R^2/2}}{\pi} \left[ \frac{\Delta\theta_i}{2} - \sum_{k=0}^{K-1} U1_{K-k} J_{k+1} \right], \tag{69}$$

where

$$J_k \equiv \left(\frac{R}{\sqrt{2}}\right)^k \int_{\theta_1}^{\theta_2} \cos^k\theta\,d\theta, \quad |\theta_1| < \frac{\pi}{2}, \quad |\theta_2| < \frac{\pi}{2}, \tag{70}$$

$$\begin{cases} J_{k+1} = \dfrac{1}{k+1} \left\{ \left[ h_2 g_2^k - h_1 g_1^k \right] + \dfrac{kR^2}{2} J_{k-1} \right\} \\ J_0 = \Delta\theta, \quad J_1 = h_2 - h_1 \end{cases} \tag{71}$$

$$g_j = \frac{R}{\sqrt{2}} \cos\theta_j, \quad h_j = \frac{R}{\sqrt{2}} \sin\theta_j, \quad j = 1, 2, \tag{72}$$

$$R^2 = \bar{x}^2 + \bar{y}^2 \ (\text{vertex of } A_i \text{ at } (\bar{x}, \bar{y})). \tag{73}$$

Since $u \geqslant 0$ in (68), this requires the constraints on $\theta_1$, $\theta_2$ given in (70). For $\pi/2 < \theta \leqslant \pi$, we use, in addition to (69),

$$P(A[R, \theta]) = \frac{1}{2} \operatorname{erfc}\left(\frac{R}{\sqrt{2}} \sin \theta\right) - P(A[R, \pi - \theta]), \tag{74}$$

where $A[R, \theta]$ denotes an angular region with its vertex a distance R from the origin and with angular measure $\theta$, where one side of $A[R, \theta]$ is formed by the line L described above. See pp 13–14 of [9] for more details.

We note that since

$$\Omega = \sum_{i=1}^{N} \Delta\theta_i, \tag{75}$$

and $\Delta\theta_i$ already occurs in (69), little additional computing is necessary to obtain W in (62).

Since most of the programming is already available, POLYCV is also designed to yield the normal probability P(A1) over a single semi-infinite angular region A1.* However for polygons each of the angular measures $\Delta\theta_i$ in (75) is in the interval $(-\pi, \pi]$, whereas for the single angular region case the angular measure of A1 is always in the interval $[0, 2\pi)$.** The angular region A1 is specified by three points. The *first point is always at the vertex of A1* the second and third points are taken such that $\Delta\theta$ of A1 is $\geqslant 0$. Figures 6 and 7 show typical angular regions.

In addition to $P(= P(\Pi))$, POLCV also supplies as output the area of $\Pi$, $A(\Pi)$, where

$$A(\Pi) = S_x S_y [1 - c^2]^{1/2} A(\bar{\Pi}), \tag{76}$$

$$A(\bar{\Pi}) = \sum_{i=1}^{N} \bar{x}_i(\bar{y}_{i+1} - \bar{y}_{i-1}), \quad \bar{y}_0 \equiv \bar{y}_N, \quad \bar{y}_{N+1} \equiv \bar{y}_1. \tag{77}$$

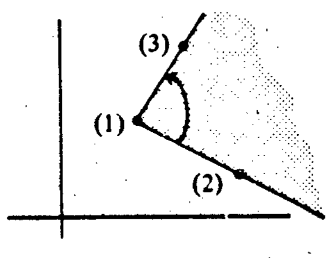



**Figure 6. Angular Region, $\Delta\theta < \pi$**

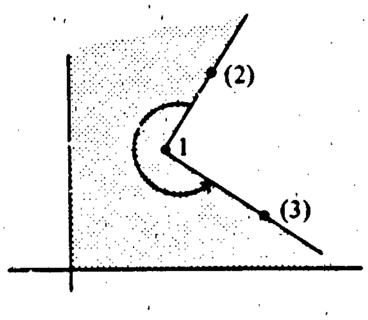



**Figure 7. Angular Region, $\Delta\theta > \pi$**

*P(A1) corresponds to a Bivariate Normal Integral. See [9, p. 18].
**Since $\Delta\theta$, in computing P(A1), is always $\geqslant 0$, P(A1) is always non-negative.

In (77), the $\bar{x}_i$, $\bar{y}_i$ refer to the coordinates of the $i$th vertex of the *transformed* polygon $\bar{\Pi}$ as obtained from (60). For PO (NO) specified polygons $A(\Pi)$ is always $>(<)0$. Thus, as indicated earlier for arbitrary polygons, one should call $A(\Pi)$ the "signed" area of $\Pi$.

The input for POLYCV is specified in data statements accordingly:

HP-9845: P8, P9, $M_x$, $M_y$, C, $S_x$, $S_y$, N, $x_1$, $y_1$, $x_2$, $y_2$, ..., $x_K$, $y_K$,

HP-85: P8, P9, M1, M2, C, S1, S2, N, $x_1$, $y_1$, $x_2$, $y_2$, ..., $x_K$, $y_K$.

For the HP-85, M1 = $M_x$, M2 = $M_y$, S1 = $S_x$, S2 = $S_y$.

If P8 = 0, then the vertex coordinates $x_i$, $y_i$, i = 1 to K are stored in data statements immediately following the initial data statement above. POLYCV stores them in arrays X(*), Y(*). If P8 $\neq$ 0, it is assumed the vertex coordinates are already stored in arrays X(*), Y(*) which is convenient if the vertices are machine generated.

P9: Determines the output desired.

- P9 = 0. No listing of the $x_i$, $y_i$; no plot of $\Pi$.
- P9 $\geq$ 1. A listing of the $x_i$, $y_i$ is printed.
- P9 > 1 or P9 < 0. A plot of $\Pi$ or angular region A1 in the xy-plane, depending on N, is given on the CRT and dumped onto the printer.

N: If N $\geq$ 3, then K is set to N by POLYCV and $P(\Pi)$ is found.

If N = 1, then K is set to 3 and P(A1) is found.

$x_i$, $y_i$, i = 1, 2, ..., K: ($x_i$, $y_i$) denotes the $i$th point specifying a polygon $\Pi$ or an angular region A1. If $\Pi$ is simple the points should be ordered so that $\Pi$ is positively oriented. In the case of an angular region A1(N = 1), ($x_1$, $y_1$) locates the vertex of A1, points ($x_2$, $y_2$) and ($x_3$, $y_3$) are ordered so that one rotates from ($x_2$, $y_2$) to ($x_3$, $y_3$) in a counterclockwise direction about ($x_1$, $y_1$).

Four output quantities are always printed. They are P, A, W, I1.

P: Contains the value of $P(\Pi)$ if N $\geq$ 3 or P(A1) if N = 1.

A: Contains the area of $\Pi$ if N $\geq$ 3 or is set to 0 if N = 1.

W: Denotes the "winding number of $\Pi$," see page 18 of [11]. For a simple (PO) polygon it is always one. It is set to zero if N = 1.

I1: If I1 = 0 or 2 output is acceptable, I1 = 2 indicates that two or more consecutive sides of $\Pi$ overlap. If I1 = 1, then angular region A1 with N = 1 is not well-defined, i.e., the vertex of A1 and at least one of the other two points specifying A1 are too close to each other. If I1 = −1 then A1 may not be well-defined – the angular measure of A1 is close to 0 or $2\pi$. A value for P is given. If I1 = 3, then c, the correlation coefficient, does not satisfy $c^2 < 1$ and is unacceptable.

The routine is presently set to yield P(Π) or P(A1) to approximately 9-decimal-digit accuracy. The computing time can be reduced significantly by requiring less accuracy. It is not difficult to modify the program to do this. The necessary changes are indicated in [11].

The final program MLEQRE provides, from quantal response experiments, the maximum likelihood estimates $\mu$, $\sigma$ for the mean $\mu_0$ and the standard deviation $\sigma_0$ of a normal distribution. It also makes available the covariance matrix elements and a plot of elliptical confidence regions.

MLEQRE is based on the development given in [7], [8]. It uses independent variables $\alpha$ and $\beta$ instead of u and s of (9)–(11), where

$$\alpha = u/s, \quad \beta = 1/s > 0. \tag{78}$$

It achieves the maximization of F, where F is given by (9), by maximizing the logarithm of F over $\alpha$ and $\beta$, where

$$L(\alpha, \beta) \equiv \ln F = \sum_{i=1}^{N} c_i \ln P(s_i) + \sum_{j=1}^{M} d_j \ln Q(t_j), \tag{79}$$

with $Z(\cdot)$, $P(\cdot)$, $Q(\cdot)$ defined in (10) and with

$$\begin{cases} s_i = \beta a_i - \alpha, \quad t_j = \beta b_j - \alpha \\ c_i \equiv \text{the number of times } a_i \text{ occurs.} \\ d_j \equiv \text{the number of times } b_j \text{ occurs.} \\ \bar{N} = \sum_{i=1}^{N} c_i, \quad \bar{M} = \sum_{j=1}^{M} d_j \end{cases} \tag{80}$$

The $a_i$ and $b_j$ are input stimuli associated with successful and unsuccessful tests, respectively. Consequently, the $a_i$ are called "successes" and the $b_j$ are called "failures." They may take any real values, with $\bar{N}$ denoting the *total* number of successes and $\bar{M}$ the *total* number of failures. In order to take advantage of the situation where repeated values of the $a_i$ and/or $b_j$ occur, (79) is written in terms of N and M, rather than $\bar{N}$ and $\bar{M}$, where N and M denote the number of *different* $a_i$ and $b_j$, respectively. This is an important feature if repeated values occur, because, by (9) and (79), the computation time to determine $\mu$ and $\sigma$ will be proportional to N + M rather than $\bar{N} + \bar{M}$.

The values of $\alpha$ and $\beta$ that maximize L (and also F) are denoted by A and B, respectively. Hence

$$A = \mu/\sigma, \quad B = 1/\sigma > 0. \tag{81}$$

The maximization is achieved by using the Newton-Raphson (N-R) procedure in two independent variables Initial estimates for A and B, which are required for (N-R), are denoted by A0 and B0. Either they are supplied by the user, or with

$$\sum_1 = \frac{1}{\bar{N}} \sum_{i=1}^{N} c_i a_i, \quad \sum_2 = \frac{1}{\bar{M}} \sum_{j=1}^{M} d_j b_j, \quad \sum_3 = \frac{1}{(\bar{N}+\bar{M})}\left( \sum_{i=1}^{N} c_i a_i^2 + \sum_{j=1}^{M} d_j b_j^2 \right)$$

MLEQRE uses

$$\begin{cases} B0 = \left\{ \sum_3 - \left[ \frac{1}{(\bar{N}+\bar{M})} \left( \bar{N}\,\Sigma_1 + \bar{M}\,\Sigma_2 \right) \right]^2 \right\}^{-1/2} > 0 \\ A0 = \frac{1}{2}\,(\Sigma_1 + \Sigma_2) \cdot B0. \end{cases} \tag{82}$$

The (N-R) corrections D1, D2*, beginning with corrections to A0, B0, are found by solving two linear equations (see (55) of [7] and/or (4.1) of [8] with their coefficients expressed in terms of the first and second partial derivatives of L with respect to $\alpha$ and $\beta$ (see (115)–(119) of [7] and/or (3.6)–(3.10) of [8]). For completeness, we give these relationships here again.

$$\begin{cases} (D1)L_{\alpha\alpha} + (D2)L_{\alpha\beta} = -L_\alpha \\ (D1)L_{\alpha\beta} + (D2)L_{\beta\beta} = -L_\beta \end{cases} \quad \left[ \text{(N-R) Equations, } L_{\alpha\alpha} \equiv \frac{\partial^2 L}{\partial \alpha^2} \right] \tag{83}$$

$$\begin{cases} D1 = (L_\beta L_{\alpha\beta} - L_\alpha L_{\beta\beta})/\Delta, \quad D2 = (L_\alpha L_{\alpha\beta} - L_\beta L_{\alpha\alpha})/\Delta \\ \Delta = L_{\alpha\alpha} L_{\beta\beta} - L_{\alpha\beta}^2 > 0 \end{cases} \tag{84}$$

$$L_\alpha = \sum_{j=1}^{M} d_j y_j / Q_j - \sum_{i=1}^{N} c_i x_i / P_i, \quad y_j \equiv Z(t_j), \quad x_i \equiv Z(s_i) \tag{85}$$

$$L_\beta = \sum_{i=1}^{N} c_i a_i x_i / P_i - \sum_{j=1}^{M} d_j b_j y_j / Q_j \tag{86}$$

$$L_{\alpha\alpha} = -\sum_{j=1}^{M} d_j (y_j/Q_j)[(y_j/Q_j) - t_j] - \sum_{i=1}^{N} c_i (x_i/P_i)[(x_i/P_i) + s_i] < 0 \tag{87}$$

---

*Some quantities such as D1 and D2 are used both as real variables and also as BASIC variables which contain the corresponding real variable values. It should be clear from the context which is intended.

$$L_{\alpha\beta} = \sum_{i=1}^{N} c_i a_i (x_i/P_i)[(x_i/P_i) + s_i] + \sum_{j=1}^{M} d_j b_j (y_j/Q_j)[(y_j/Q_j) - t_j] \tag{88}$$

$$L_{\beta\beta} = -\sum_{j=1}^{M} d_j b_j^2 (y_j/Q_j)[(y_j/Q_j) - t_j] - \sum_{i=1}^{N} c_i a_i^2 (x_i/P_i)[(x_i/P_i) + s_i] < 0. \tag{89}$$

The equation for a confidence ellipse in the us-plane with center at $(\mu, \sigma)$ is given by

$$A_{uu}(u-\mu)^2 + 2A_{us}(u-\mu)(s-\sigma) + A_{ss}(s-\sigma)^2 = \chi^2_{1-\gamma}, \tag{90}$$

where $\chi^2_{1-\gamma}$ is obtained from a chi-squared table with two degrees of freedom; $\gamma$ denotes the probability that the ellipse contains $(\mu_0, \sigma_0)$. For $\gamma = .5$, $\chi^2_{.5} = 1.39$, and for $\gamma = .95$, $\chi^2_{.05} = 5.99$. See page 42 of [7] for other values. The coefficients in (90), which make up the elements of the inverse of the covariance matrix, are available directly from

$$\left\{\begin{aligned} \sigma^2 A_{uu} &= \sum_{i=1}^{N} c_i (x_i/P_i)(x_i/Q_i) + \sum_{j=1}^{M} d_j (y_j/P_j)(y_j/Q_j) \\ \sigma^2 A_{us} &= \sum_{i=1}^{N} c_i s_i (x_i/P_i)(x_i/Q_i) + \sum_{j=1}^{M} d_j t_j (y_j/P_j)(y_j/Q_j) \\ \sigma^2 A_{ss} &= \sum_{i=1}^{N} c_i s_i^2 (x_i/P_i)(x_i/Q_i) + \sum_{j=1}^{M} d_j t_j^2 (y_i/P_j)(y_i/Q_j), \end{aligned}\right. \tag{91}$$

which are evaluated at $\alpha = \mu/\sigma$ and $\beta = 1/\sigma$; it is recalled from (78), (80) that

$$\left\{\begin{aligned} s_i &= a_i\beta - \alpha \\ t_j &= b_j\beta - \alpha, \quad \alpha \equiv u/s, \quad \beta = 1/s > 0. \end{aligned}\right.$$

Derivations of (90) and (91) are given in [7].

The evaluation of the quantities in (84)–(90) requires an efficient and high precision subroutine to compute

$$Y \equiv y/Q, \quad Y1 \equiv (y/Q)[y/Q - t], \quad Y2 \equiv (y/Q)(y/P), \tag{92}$$

where the subscript j has been dropped. It is easy to show that the corresponding quantities in terms of $(x_i/P_i)$ can be obtained from the same subroutine by changing the sign of the input argument, i.e., changing $s_i$ to $-s_i$.

Cody's rational approximations for the complementary error function as given by (12)–(15) are fundamental to the evaluation of the quantities in (92). Using J = 4, J1 = 6 and J2 = 4 in (12)–(15) yields a minimum of eleven and one-half *significant* digits of accuracy for the complementary error function, erfc (·).

For completeness, we give the expressions used for Y, Y1, Y2, where we use the facts that

$$\begin{cases} y(t) = x(t) = x(-t) = y(-t) \\ P(t) = Q(-t). \end{cases} \tag{93}$$

Let $t = t_j$ or $(-s_i)$, $K1 = t/\sqrt{2}$, $C2 = \sqrt{2/\pi}$, $E = \exp(-K1^2)$. Also let

$$\Sigma = \Sigma_N/\Sigma_D,$$

where $\Sigma_N$ denotes the numerator sum in (12), (13) or (14) and $\Sigma_D$ denotes the denominator sum. For example if $1/2 < K1 \leq 4$, then, referring to (13),

$$\sum_N = \sum_{i=0}^{J1-1} R9(J1-i)(K1)^i.$$

Now for:

$\underline{|K1| \leq 1/2}$

$$\begin{cases} Y = (C2)E/\text{erfc}(K1) \\ Y1 = Y(Y-t) \\ Y2 = (C2)EY/(1+K1\,\Sigma) \end{cases} \tag{94}$$

$\underline{-4 \leq K1 < -1/2}$

$$\begin{cases} Y = (C2)E/(2-E\,\Sigma) \\ Y1 = Y(Y-t) \\ Y2 = (C2)Y/\Sigma \end{cases} \tag{95}$$

$\underline{1/2 < K1 \leq 4}$

$$\begin{cases} Y = C2/\Sigma \\ Y1 = Y(Y-t) \\ Y2 = (C2)EY/(2-E\Sigma) \end{cases} \tag{96}$$

$\underline{-5.5 < K1 \le -4}$

$$\begin{cases} Y = (C2)E/[2 - \mathrm{erfc}(\overline{K1})], \quad \overline{K1} \equiv |K1| \\ Y1 = Y(Y - t) \\ Y2 = (C2)\overline{K1}\, Y/(1/\sqrt{\pi} + \Sigma/K1^2) \end{cases} \tag{97}$$

$\underline{K1 > 4}$

$$\begin{cases} Y = (C2)K1/[(1/\sqrt{\pi}) + \Sigma/K1^2] \\ Y1 = -(2/\sqrt{\pi})\,\Sigma/[(1/\sqrt{\pi}) + \Sigma/K1^2]^2 \\ Y2 = (C2)EY/[2 - \mathrm{erfc}(K1)] \end{cases} \tag{98}$$

$\underline{K1 < -5.5}$

$$Y = Y1 = Y2 = 0 \quad (Y < 3 \times 10^{-14}) \tag{99}$$

The subroutine for computing Y, Y1, Y2 using the above begins on line 1215 for the HP 9845 program and on line 695 for the HP-85 program. It is the core of MLEQRE.

There are numerous BASIC variables in MLEQRE which are pertinent; we list them here for convenience.

A(i) : contains the *i*th listed success value $a_i$, $(1 \le i \le N)$.

B(j) : contains the *j*th listed failure value $b_j$ $(1 \le j \le M)$.

C(i) : contains $c_i$, the number of $a_i$ at a fixed i.

D(j) : contains $d_j$, the number of $b_j$ at a fixed j.

N : contains the number of *different* $a_i$.

M : contains the number of *different* $b_j$.

A0 : contains the current $\alpha(= u/s)$ value.

B0 : contains the current $\beta(= 1/s)$ value.

U : contains $\mu$.

S : contains $\sigma$.

A1 : contains the initial estimate for $\alpha$ upon EXIT

B1 : contains the initial estimate for $\beta$ upon EXIT

L0 : $\begin{cases} \text{If L0 contains one, then } c_i \text{ and } d_j \text{ are one for all i and j.} \\ \text{If L0 does not contain one, then some } c_i \text{ and/or } d_j \text{ may be larger than one.} \end{cases}$

D1 : contains the current (N-R) correction D1 to A0.

D2 : contains the current (N-R) correction D2 to B0.

L1 : contains $L_\alpha (\equiv \partial L/\partial\alpha)$

L2 : contains $L_\beta$

L3 : contains $L_{\alpha\alpha}$

L4 : contains $L_{\alpha\beta}$

L5 : contains $L_{\beta\beta}$

L6, L7, L8 : contain the confidence ellipse coefficients $A_{uu}$, $A_{us}$, $A_{ss}$, respectively

L2, L3, L4 : change their contents, after completion of the (N-R) procedure, to the covariance matrix elements $A^{uu}$, $A^{us}$, $A^{ss}$, respectively.

Y : contains either $(x_i/P_i)$ or $(y_j/Q_j)$

Y1 : contains either $(x_i/P_i)[(x_i/P_i) + s_i]$ or $(y_j/Q_j)[(y_j/Q_j) - t_j]$

Y2 : contains either $(x_i/P_i)(x_i/Q_i)$ or $(y_j/P_j)(y_j/Q_j)$

Z : contains 0, 1, 2 and is used to signal that one cycle of (N-R) remains to be carried out. This allows Y2 to be computed, for use in the evaluation of the confidence ellipse coefficients in (90), only on the last (N-R) iteration.

The input data for MLEQRE is stored in data statements. If L0 $\neq$ 1, then data is stored sequentially in the following variables: N, M, L0, A0, B0, P8, C(1), A(1), C(2), A(2), ..., C(N), A(N), D(1), B(1), D(2), B(2), ..., D(M), B(M). If L0 = 1, then data is stored sequentially in the following variables: N, M, L0, A0, B0, P8, A(1), A(2), ..., A(N), B(1), B(2), ..., B(M); the arrays C(i) and D(j) are stored with ones by MLEQRE in this case.

A0 and B0 contain initial estimates of A and B (see (82)), supplied by the user. If, however, B0 contains zero then MLEQRE supplies the initial estimates.

The plotting of the confidence ellipses at the 50 and 95% levels begins at line 1440 for the HP-9845 and at line 830 for the HP-85. If P8 $\geq$ 2, then plot appears on CRT and also the printer. If P8 = 1, then plot appears only on the CRT. If P8 = 0, then no plot is constructed.

The following output is given with the format differing slightly between the HP-9845 and HP-85.

Values in N, M, L0, initial values in A0, B0, P8.

Values of $c_i$, $a_i$ and values of $d_j$, $b_j$ if L0 $\neq$ 1, otherwise values of $a_i$ and $b_j$ only.

The maximum likelihood estimates MU($=\mu$), SIG($=\sigma$), covariance matrix elements, initial values for A($=\mu/\sigma$), B($= 1/\sigma$).

Final values of A0, B0.

Final values of D1, D2 ((N-R) corrections).

Number of (N-R) inerations.

In [7], necessary and sufficient conditions were derived for the first time that assure the existence and moreover the uniqueness of the maximum likelihood estimates $\mu$, $\sigma$. They demand that

$$\begin{cases} \max\limits_{j} b_j > \min\limits_{i} a_i , \\ \frac{1}{N} \sum\limits_{i=1}^{N} c_i a_i > \frac{1}{M} \sum\limits_{j=1}^{M} d_j b_j . \end{cases} \tag{99}$$

If either of these inequalities is not satisfied, a message is printed and MLEQRE terminates – for the given stimuli, $\mu$ and $\sigma$ do not exist.

A Fortran IV program based on [7] and upon which MLEQRE is modeled has been available for sometime on the CDC 6700. A diluted version of that program is available in BASIC for the 4051–4054 series Textronix desk-top computers, [13]. It does not contain the plotting feature for confidence ellipses, and it does not have the capability to take advantage of the increased efficiency when stimuli are repeated, and in general it does not appear to be as efficient as MLEQRE. The program is very difficult to follow and we have not been able to verify its correctness.

## III. LISTINGS OF PROGRAMS AND SAMPLE OUTPUTS

A short summary of the input and output associated with a particular program, including examples, is given starting with CIRCV. This is followed by the HP-9845 listing for that particular program and then the corresponding HP-85 listing.

It is assumed in operating the HP-9845 programs that

PRINTER IS 0

has been executed.

The BASIC language for the HP-85 allows for multi-statement lines, where the statements on a numbered line are separated by the symbol @. Care must be taken in interpreting a multi-statement line when an IF ··· THEN or an IF ··· THEN ··· ELSE statement is not the last statement in the line. In the first case, when the IF-statement is *true* and THEN is not followed by a program transfer, such as a GO TO, the execution of the IF-statement is followed by executing the *next* statement of the *same* line. When the IF-statement is *false*, the program proceeds directly to the next sequentially numbered line. For example:

```
100: IF A = B THEN C = B @ GO TO 500
110:
```

If A = B, then at line 100 C is set to B and execution continues at line 500. If A ≠ B at line 100, then execution continues at line 110.

In the case of the IF ··· THEN ··· ELSE statement, if the IF part is *true*, and THEN is not followed by a program transfer, then the program proceeds to the next numbered line. If, however, the IF part of the IF ··· THEN ··· ELSE statement is *false*, then execution of the ELSE part of the statement is followed by executing the next statement of the *same* line. For example:

```
100: IF A = B THEN C = B ELSE D = B @ GO TO 500
110: ···
```

If A = B, then C is set to B and program proceeds to line 110; if A ≠ B, then D is set to B and program proceeds to line 500.

## CIRCV or GCEF

*Input:* R, D, V

CIRCV: $R = \bar{R}/\sigma_x, \quad D = \sqrt{x^2 + y^2}/\sigma_x, \quad V = 1.$

GCEF: $R = K \equiv \bar{R}/\sigma_x, \quad D = c \equiv \sigma_y/\sigma_x < 1, \quad V = 0.$

If R and/or D < 0, then P set to (−1). Input unacceptable.

If D > 1 for GCEF, then P set to (−1). Input unacceptable.

*Output:*

CIRCV: $P = P(R, D), \quad S2 = \frac{1}{R}\frac{\partial P}{\partial R}$

GCEF: $P = F(K, c), \quad S2 = (D/K)\partial F/\partial K, \quad D \neq 0$

$= \sqrt{\pi/2}\ \partial F/\partial K, \quad D = 0$

Accuracy: 6 decimal-digits for P and S2

| Case | EXAMPLES | | | | Input | | | Output | |
|---|---|---|---|---|---|---|---|---|---|
| | $\bar{R}$ | $\sqrt{x^2 + y^2}$ | $\sigma_x$ | $\sigma_y$ | R | D | V | P | S2 |
| ① | 3 | 4 | 2 | 2 | 1.5 | 2 | 1 | .209232218046 | .214447041618 |
| ② | 6 | 4 | 2 | 2 | 3 | 2 | 1 | .785637894167 | .101082811001 |
| ③ | 6 | 5 | 2 | 2 | 3 | 2.5 | 1 | .623010408440 | .130887896597 |
| ④ | 5 | 6 | 2 | 2 | 2.5 | 3 | 1 | .246101694960 | .130887896597 |
| ⑤ | 2 | 0 | 2 | 4 | .5 | .5 | 0 | .215288716030 | .738059987125* |
| ⑥ | 8 | 0 | 2 | 1 | 4 | .5 | 0 | .999926141520 | 3.90547542875(−5) |
| ⑦ | 2 | 0 | 1 | 0 | 2 | 0 | 0 | .954499736146 | .135335283238 |
| ⑧ | 2 | 0 | 1 | 1 | 2 | 1 | 0 | .864664716770 | .135335283237 |

*$\sigma_x$ and $\sigma_y$ interchanged so that $\sigma_y/\sigma_x < 1$.

Output values above are for the HP-85. The corresponding values for the HP-9845B differ in the last one to three digits.

CIRCV – HP 9845

```
LIST
100    ! THIS PROGRAM IS CALLED "CIRCV". IT SUPPLIES TWO
         FUNCTIONS: P(R,D), THE CIRCULAR COVERAGE FUNCTION
105    ! OR F(K,C),THE GENERALIZED CIRCULAR ERROR FUNCTION.
110    ! THE INPUT IS R,D,V, WHERE IF V=0 THEN K=R AND C=D. IF V#0
         THE OUTPUT IS P=P(R,D); IF V=0 THEN THE OUTPUT IS P=F(K,C).
115    ! INPUT R OR D <0 NOT PERMITTED. ALSO FOR V=0, ABS(D-.5)>.5
         NOT ALLOWED. IN SUCH CASES P SET TO -1.
120    ! LET Pr DENOTE THE PARTIAL DERIVATIVE OF P WITH
         RESPECT TO R. THEN Pr=R*S2.
125    ! LET Fk DENOTE THE PARTIAL DERIVATIVE OF F(K,C)
         WITH RESPECT TO K. THEN Fk=(K/C)*S2, C#0.
130    ! IF C=0 THEN Fk= SQR(2/PI)*S2. S2 IS AVAILABLE
         INTERNALLY.
135    ! SOURCES: MATH OF COMP APRIL 1961,PP169,173 AND OCT.
         1961, PP 375,382. NWL REPORT #1768, JAN. 1962.
140    ! IEEE TRANS. INFO. TH. APRIL 1965, P. 312.
145    ! PROGRAM IS SET FOR SIX DECIMAL DIGIT ACCURACY.
150    P9(3)=21.3853322378
155    P9(2)=1.72227577039
160    P9(1)=.316652890658
165    Q9(2)=18.9522572415
170    Q9(1)=7.8437457083
175    R9(5)=7.3738883116
180    R9(4)=6.8650184849
185    R9(3)=3.0317993362
190    R9(2)=.56316961891
195    R9(1)=4.3187787405E-5
200    S9(4)=7.3739608908
205    S9(3)=15.18490819
210    S9(2)=12.795529509
215    S9(1)=5.3542167949
220    B1=.0090005
225    S2=0
230    P=0
235    B2=.707106781187
240    IF (R>=0) AND (D>=0) THEN 255
245    P=-1
250    RETURN
255    IF R=0 THEN 250
260    IF V=0 THEN 605   ! COMPUTE GCEF.
265    A1=R-D
270    A=ABS(A1)
275    IF A<5.386773 THEN 295
280    IF A1<0 THEN 250
285    P=1
290    RETURN
295    T=R*D
300    T3=.5*R*R
305    B=.5*D*D
310    N=0
```

```
315   IF T>7 THEN 455
320   T1=B2*T-1
325   T2=T3*B
330   S0=EXP(-T3-B)
335   S1=EXP(-B)
340   IF T3>.0005 THEN 355
345   S1=S1*T3
350   GOTO 360
355   S1=S1-S0
360   S2=S0
365   T0=S1
370   N=N+1
375   M=1/N
380   S0=T2*M*M*S0
385   T0=B*M*T0-S0
390   S1=S1+T0
395   S2=S2+S0
400   IF T1>N THEN 370
405   IF T0>B1 THEN 420
410   P=S1
415   RETURN
420   N=N+1
425   M=1/N
430   S0=T2*M*M*S0
435   T0=B*M*T0-S0
440   S1=S1+T0
445   S2=S2+S0
450   GOTO 405
455   T1=2*ABS(T3-B)
460   A=A*B2
465   T3=1/(T+T)
470   T2=SQR(T3)
475   S1=.5*A1*A1
480   S2=EXP(-S1)
485   S0=.564189583545*T2*S2
490   GOSUB 710
495   T0=(R+D)*B2*T2*E
500   T2=S1*T3
505   T3=.5*T3
510   S1=T0
515   S2=S0
520   N=N+2
525   M=N-1
530   A=M/N
535   S0=A*T3*S0
540   T0=T1*S0-T2*A*T0
545   S0=M*S0
550   S1=S1+T0
555   S2=S2+S0
560   IF T0-B1>0 THEN 520
565   IF S0-B1>0 THEN 580
570   P=.5*ABS(1+SGN(A1)-S2-SGN(A1)*S1)
575   RETURN
```

```
580   N=N+2
585   M=N-1
590   S0=M*M*T3*S0/N
595   S2=S2+S0
600   GOTO 565
605   IF ABS(D-.5)>.5 THEN 245 !START FOR GCEF.
610   IF R>=5.386773 THEN 285
615   IF D<>0 THEN 650
620   A=B2*R
625   S1=A*A
630   S2=EXP(-S1)
635   GOSUB 710
640   P=1-E
645   RETURN
650   K=R
655   C=D
660   T=.5/C
665   R=K*(1+C)*T
670   D=K*(1-C)*T
685   GOSUB 265
690   R=K
695   D=C
700   P=ABS(P+P+S2-1)
705   RETURN
710   IF ABS(A)>.5 THEN 725  !E=ERFC(A)
715   E=1-A*((P9(1)*S1+P9(2))*S1+P9(3))/((S1+Q9(1))*S1+Q9(2))
720   RETURN
725   E=((((R9(1)*A+R9(2))*A+R9(3))*A+R9(4))*A+R9(5))/((((A+S9(1))*A+S9(2)
      (3))*A+S9(4))*S2
730   RETURN
```

CIRCV – HP 85

```
100 ! THIS PROGRAM IS CALLED "CI
    RCV". IT SUPPLIES TWO FUNCTI
    ONS: P(R,D), THE CIRCULAR CO
    VERAGE
105 ! FUNCTION OR F(K,C) THE GEN
    ERALIZED CIRCULAR ERROR FUNC
    TION.
110 ! THE INPUT IS R,D,V, WHERE
    IF V=0 THEN K=R AND C=D. IF
    V#0 THE OUTPUT IS P=P(R,D);
    IF V=0
115 ! THEN THE OUTPUT IS F(K,C).
    INPUT R OR D<0 NOT PERMITTED
    .ALSO FOR V=0 ABS(D-.5)>.5 N
    OT
120 ! ALLOWED. IN SUCH CASES P S
    ET TO -1.
125 ! LET Pr=THE PARTIAL DERIVAT
    IVE OF P WITH RESPECT TO R.
    THEN Pr=R*S2.
130 ! LET Fk= THE PARTIAL DERIVA
    TIVE OF F WITH RESPECT TO K.
    IF  C#0 THEN Fk=(K/C)*S2.
135 ! IF C=0 THEN Fk=SQR(2/PI)*S
    2. S2 IS AVAILABLE INTERNALLY
140 ! PROGRAM IS SET FOR 6-DIGIT
    ACCURACY.
145 ! SOURCES: MATH OF COMP. APR
    IL,1961,PP.169-173 AND OCT.
    1961, PP. 375-382.
150 ! SOURCES: NWL REPORT#1768,J
    AN.1962. IEEE TRANS. INFO. T
    H. APRIL,1965, P.312.
155 P9(3)=21.3853322378 @ P9(2)=
    1.72227577039 @ P9(1)=.31665
    2890658
160 Q9(2)=18.9522572415 @ Q9(1)=
    7.8437457083
165 R9(5)=7.3738883116 @ R9(4)=6
    .8650184849 @ R9(3)=3.031799
    3362 @ R9(2)=.56316961891
170 R9(1)=4.3187787405E-5 @ S9(4
    )=7.3739608908 @ S9(3)=15.18
    490819 @ S9(2)=12.795529509
175 S9(1)=5.3542167949
180 B1=.0000005 @ S2=0 @ P=0 @ E
    =0 @ B2=.707106781187
185 IF R>=0 AND D>=0 THEN 195
190 P=-1 @ RETURN
195 IF R=0 THEN RETURN
200 IF V=0 THEN 325
205 A1=R-D @ A=ABS(A1) @ IF A<5.
    386773 THEN 220
210 IF A1<0 THEN P=0 ELSE P=1
215 RETURN
220 T=R*D @ T3=.5*R*R @ B=.5*D*D
     @ N=0
225 IF T>7 THEN 275 ELSE T1=B2*T
    -1
230 T2=T3*B @ S0=EXP(-T3-B) @ IF
     T3>.0005 THEN S1=EXP(-B)-S0
     ELSE S1=EXP(-B)*T3
235 S2=S0 @ T0=S1
240 N=N+1 @ M=1/N
245 S0=T2*M*M*S0 @ T0=B*M*T0-S0
250 S1=T0+S1 @ S2=S2+S0
255 IF T1>N THEN 240
260 IF T0>B1 THEN 270
265 P=S1 @ RETURN
270 N=N+1 @ M=1/N @ S0=T2*M*M*S0
     @ T0=B*M*T0-S0 @ S1=T0+S1 @
     S2=S2+S0 @ GOTO 260
275 T1=2*ABS(T3-B) @ A=A*B2 @ T3
    =1/(T+T) @ T2=SQR(T3) @ S1=.
    5*A1*A1
280 S2=EXP(-S1) @ S0=.5*T2*1.128
    37916709*S2 @ T0=(R+D)*B2*FN
    E(A)*T2
285 T2=S1*T3 @ T3=.5*T3 @ S1=T0
    @ S2=S0
290 N=N+2 @ M=N-1 @ A=M/N
295 S0=A*T3*S0 @ T0=T1*S0-T2*A*T
    0 @ S0=M*S0
300 S1=S1+T0 @ S2=S2+S0
305 IF T0-B1>0 THEN 290
310 IF S0-B1>0 THEN 320
315 P=.5*ABS(1+SGN(A1)-S2-SGN(A1
    )*S1) @ RETURN
320 N=N+2 @ M=N-1 @ S0=M*M*T3*S0
    /N @ S2=S2+S0 @ GOTO 310
325 IF ABS(D-.5)<=.5 THEN 335
330 P=-1 @ RETURN
335 IF R<5.386773 THEN 345
340 P=1 @ RETURN
345 IF D#0 THEN 355
350 A=B2*R @ S1=A*A @ S2=EXP(-S1
    ) @ P=1-FNE(A) @ RETURN
355 K=R @ C=D @ T=.5/C @ R=K*(1+
    C)*T @ D=K*(1-C)*T
360 GOSUB 205
365 R=K @ D=C @ P=ABS(P+P+S2-1)
370 RETURN
375 DEF FNE(A) ! FNE(A)=ERFC(A)
380 IF ABS(A)>.5 THEN 395
385 FNE=1-A*((P9(1)*S1+P9(2))*S1
    +P9(3))/((S1+Q9(1))*S1+Q9(2)
    )
390 GOTO 400
395 FNE=((((R9(1)*A+R9(2))*A+R9(
    3))*A+R9(4))*A+R9(5))/((((A+
    S9(1))*A+S9(2))*A+S9(3))*A+S
    9(4))*S2
400 FN END
```

**ELLCV or ELLCV3**

*Input:* $\bar{R}, \bar{H}, \bar{K}$, S1, S2

$(\bar{H} = \bar{h}, \bar{K} = \bar{k}, S1 = \sigma_x, \; S2 = \sigma_y)$

*Output:* $P = P(\bar{R}, \bar{h}, \bar{k}, \sigma_x, \sigma_y)$

*Accuracy:* 6-decimal-digits for ELLCV

3-decimal-digits for ELLCV3

**EXAMPLES**

1) $\bar{R} = 5, \bar{H} = 2, \bar{K} = 3, \sigma_x = S1 = 3, \sigma_y = S2 = 2$ (See Example 3 of POLYCV, page 59)

   P = .588490575749 (ELLCV)

   P = .588490579217 (ELLCV3)

2) $\bar{R} = 2, \bar{H} = 0, \bar{K} = 0, \sigma_x = 2, \sigma_y = 4$ (See Example 5 for CIRCV, page 26)

   P = .215288716038 (ELLCV)

   P = .215288754652 (ELLCV3)

3) $\bar{R} = 3, \bar{H} = 2, \bar{K} = 2\sqrt{3}, \sigma_x = 2, \sigma_y = 2$ (See Example 1 for CIRCV, page 26)

   P = .209232220601 (ELLCV)

   P = .209232478927 (ELLCV3)

4) $\bar{R} = 3, \bar{H} = 0, \bar{K} = .3, \sigma_x = 1, \sigma_y = 1/10$

   P = .997146500681 (ELLCV)

   P = .9970944451746 (ELLCV3)

NOTE: The bars on R, H, K are to conform with the discussion in the previous section. As BASIC variables the bars are deleted.

ELLCV – HP 9845

```
100   ! THIS PROGRAM IS CALLED "ELLCV". IT SUPPLIES
        THE ELLIPTICAL COVERAGE FUNCTION:P(R,H,K,S1,S2).
105   ! P DENOTES THE PROBABILITY OF A SHOT, NORMALLY
        DISTRIBUTED WITH MEAN (0,0) AND STANDARD
110   ! DEVIATIONS S1,S2 IN THE X AND Y DIRECTIONS,
        RESPECTIVELY, FALLING IN A CIRCLE IN THE
115   ! XY-PLANE OF RADIUS R AND CENTERED AT (H,K). THE
        INPUT IS R,H,K,S1,S2. THE OUTPUT IS P.
120   ! PROGRAM IS SET FOR 6-DECIMAL-DIGIT ACCURACY IN P.
125   ! ELLCV USES ERFC WITH 9 DIGIT RELATIVE ACCURACY.
130   ! SOURCES: NWL REPORT #1710, AUG.1960. MATH OF
        COMP. OCT. 1961, PP. 375,382.
135   ! INPUT R,H,K,S1,S2
140   ! "ELLCV" CONSTRUCTED IN COLLABORATION WITH ALFRED MORRIS.
145   OPTION BASE 1
150   DIM P9(3),Q9(2),R9(5),S9(4),X(43),Y(43)
155   P9(1)=3.16652890658E-1
160   P9(2)=1.72227577039
165   P9(3)=21.3853322378
170   Q9(1)=7.8437457083
175   Q9(2)=18.9522572415
180   R9(1)=4.3187787405E-5
185   R9(2)=.56316961891
190   R9(3)=3.0317993362
195   R9(4)=6.8650184849
200   R9(5)=7.3738883116
205   S9(1)=5.3542167949
210   S9(2)=12.795529509
215   S9(3)=15.18490819
220   S9(4)=7.3735608908
225   X(1)=.238619186083
230   X(2)=.661209386466
235   X(3)=.932469514203
240   X(4)=.183434642496
245   X(5)=.525532409916
250   X(6)=.796666477414
255   X(7)=.960289856498
260   X(8)=.125233408511
265   X(9)=.367831498998
270   X(10)=.587317954287
275   X(11)=.769902674194
280   X(12)=.90411725637
285   X(13)=.981560634247
290   X(14)=9.50125098376E-2
295   X(15)=.281603550779
300   X(16)=.458016777657
305   X(17)=.617876244403
310   X(18)=.755404408355
315   X(19)=.865631202388
320   X(20)=.944575023073
325   X(21)=.989400934992
```

```
330    X(22)=7.65265211335E-2
335    X(23)=.227785851142
340    X(24)=.373706088715
345    X(25)=.510867001951
350    X(26)=.636053680727
355    X(27)=.74633190646
360    X(28)=.839116971822
365    X(29)=.912234428251
370    X(30)=.963971927278
375    X(31)=.993128599185
380    X(32)=6.40568928626E-2
385    X(33)=.191118867474
390    X(34)=.315042679696
395    X(35)=.433793507626
400    X(36)=.545421471389
405    X(37)=.648093651937
410    X(38)=.740124191579
415    X(39)=.820001985974
420    X(40)=.886415527004
425    X(41)=.938274552003
430    X(42)=.974728555971
435    X(43)=.995187219997
440    Y(1)=.467913934573
445    Y(2)=.360761573048
450    Y(3)=.171324492379
455    Y(4)=.362683783378
460    Y(5)=.313706645878
465    Y(6)=.222381034453
470    Y(7)=.10122853629
475    Y(8)=.249147045813
480    Y(9)=.233492536538
485    Y(10)=.203167426723
490    Y(11)=.160078328543
495    Y(12)=.106939325995
500    Y(13)=4.71753363865E-2
505    Y(14)=.189450610455
510    Y(15)=.182603415045
515    Y(16)=.169156519395
520    Y(17)=.149595988817
525    Y(18)=.124628971256
530    Y(19)=9.51585116825E-2
535    Y(20)=6.22535239386E-2
540    Y(21)=2.71524594118E-2
545    Y(22)=.152753387131
550    Y(23)=.149172986473
555    Y(24)=.142096109318
560    Y(25)=.131688638449
565    Y(26)=.118194531962
570    Y(27)=.10193011981
575    Y(28)=8.32767415767E-2
580    Y(29)=6.26720483341E-2
```

**ELLCV – HP 9845**

```
585    Y(30)=4.06014298004E-2
590    Y(31)=1.76140071392E-2
595    Y(32)=.127938195347
600    Y(33)=.125837456347
605    Y(34)=.121670472928
610    Y(35)=.115505668054
615    Y(36)=.107444270116
620    Y(37)=9.76186521041E-2
625    Y(38)=.086190161532
630    Y(39)=7.33464814111E-2
635    Y(40)=5.92985849154E-2
640    Y(41)=4.42774388174E-2
645    Y(42)=2.85313886289E-2
650    Y(43)=.0123412298
655    A=4.892
660    A1=5.387
665    A2=3.8775
670    A3=5.16
675    B1=.564189583548   ! 1/SQR(PI)
680    B=1.41421356237    ! SQR(2)
685    B2=29.019769       !B2=A1*A1
690    P=0
695    Z3=.000001*S1*S2
700    IF R*R<=Z3 THEN RETURN
705    H2=H*H+K*K
710    D=MAX(S1,S2)
715    T=R-A1*D
720    ! PROCEED TO SEE IF P=0 OR P=1
725    IF T<0 THEN 745
730    IF T*T<H2 THEN 745
735    P=1
740    RETURN
745    H8=ABS(H)
750    K8=ABS(K)
755    IF R-H8+A*S1<=0 THEN RETURN
760    IF R-K8+A*S2<=0 THEN RETURN
765    S0=SQR(H2)
770    IF S1<>S2 THEN 790
775    H8=S0
780    K8=0
785    IF R+A*S1<=H8 THEN RETURN
790    IF S0<=R THEN 910
795    D=(S0-R)/D
800    IF R*R*EXP(-.5*D*D)>Z3 THEN 910
805    RETURN
810    IF S0<R+A1*MIN(S1,S2) THEN 910
815    IF H8*K8=0 THEN 910
820    H9=H8/S1
825    K9=K8/S2
830    D=H9*H9+K9*K9
835    IF D<=B2 THEN 910
```

```
840   Z2=R/S2
845   Q=S2/S1
850   Q1=Q*Q
855   F=Q1*H9*H9+K9*K9
860   Z1=Z2*Z2*F/D
865   Z=D-Z1-B2
870   IF Z<0 THEN 880
875   IF Z*Z-4*Z1*B2>=0 THEN RETURN
880   T1=H8*H8+Q1*K8*K8
885   Z8=B2*S1*S1*T1/H2
890   R2=R*R
895   Y=H2-R2-Z8
900   IF Y<=0 THEN 910
905   IF Y*Y-4*R2*Z8>=0 THEN RETURN
910   Z8=0!     FIND LIMITS OF INTEGRATION
915   Z=K8+A3*S2
920   H3=K8-A3*S2
925   S0=S1
930   S9=S2
935   Z=R-Z
940   D1=0
945   IF Z>=0 THEN D1=SQR(Z/R)
950   IF H3>=0 THEN 970
955   H5=0
960   E3=1-D1
965   GOTO 980
970   E3=SQR(1-H3/R)-D1
975   H5=1
980   IF Z8<>0 THEN 1020
985   Z8=1
990   F=E3
995   T=D1
1000  Z=H8+A3*S1
1005  H6=H5
1010  H3=H8-A3*S1
1015  GOTO 935
1020  IF F>=E3 THEN 1065
1025  E3=F
1030  D1=T
1035  S9=S1
1040  Z8=H8
1045  S0=S2
1050  H8=K8
1055  K8=Z8
1060  H5=H6
1065  E3=.5*E3   !  BEGIN GAUSSIAN INTEGRATION
1070  N=E3*R*(.34/S0+1/(.025*ABS(R-K8)+5*S9))
1075  Z2=R/(B*S9)
1080  R8=R/(B*S0)
1085  H8=H8/(B*S0)
1090  K8=K8/(B*S9)
```

```
1095  IF N<2.75 THEN 1115
1100  J=31
1105  N1=12
1110  GOTO 1205
1115  IF N<1.35 THEN 1135
1120  J=21
1125  N1=10
1130  GOTO 1205
1135  IF N<.75 THEN 1155
1140  J=13
1145  N1=8
1150  GOTO 1205
1155  IF N<.35 THEN 1175
1160  J=7
1165  N1=6
1170  GOTO 1205
1175  IF N<.15 THEN 1195
1180  J=3
1185  N1=4
1190  GOTO 1205
1195  J=0
1200  N1=3
1205  Z=Z3=0
1210  Y=B1*E3*R8
1215  K9=1.04E-8
1220  H9=1.9999999792
1225  G3=0
1230  M=N1+N1
1235  I=-N1
1240  IF K8=0 THEN 1415
1245  FOR L=1 TO M
1250    IF I=0 THEN I=1
1255    T=E3*(SGN(I)*X(J+ABS(I))+1)+D1
1260    T9=T*T
1265    T1=R8*(1-T9)
1270    T2=H8-T1
1275    T4=EXP(-T2*T2)
1280    IF H8<>0 THEN 1295
1285    T4=T4+T4
1290    GOTO 1310
1295    IF H5<>0 THEN 1310
1300    T2=H8+T1
1305    T4=T4+EXP(-T2*T2)
1310    IF Z<>0 THEN 1340
1315    Z1=Z2*T*SQR(2-T9)
1320    K1=K8-Z1
1325    IF ABS(K1)<A2 THEN 1350
1330    IF K1>0 THEN 1395
1335    Z=1
1340    K5=H9
1345    GOTO 1360
```

```
1350    GOSUB 1560
1355    K5=K3
1360    K1=K8+Z1
1365    IF K1<A2 THEN 1380
1370    K5=K5-K9
1375    GOTO 1390
1380    GOSUB 1560
1385    K5=K5-K3
1390    G3=G3+K5*T4*T*Y(J+ABS(I))
1395    I=I+1
1400  NEXT L
1405  P=Y*G3
1410  RETURN
1415  FOR L=1 TO M
1420    IF I=0 THEN I=1
1425    T=E3*(SGN(I)*X(J+ABS(I))+1)+D1
1430    T9=T*T
1435    T1=R8*(1-T9)
1440    T2=H8-T1
1445    T4=EXP(-T2*T2)
1450    IF H8<>0 THEN 1465
1455    T4=T4+T4
1460    GOTO 1480
1465    IF H5<>0 THEN 1480
1470    T2=H8+T1
1475    T4=T4+EXP(-T2*T2)
1480    IF Z<>0 THEN 1500
1485    K1=Z2*T*SQR(2-T9)
1490    IF K1<A2 THEN 1510
1495    Z=1
1500    K5=H9
1505    GOTO 1520
1510    GOSUB 1560
1515    K5=2*(1-K3)
1520    G3=G3+K5*T4*T*Y(J+ABS(I))
1525    I=I+1
1530    NEXT L
1535    P=Y*G3
1540    RETURN
1545    REM CODY FOR K3=ERFC(K1)--9 DIGITS
1550    ! IF K1<=-A2 THEN K3=2-2.08E-8
1555    ! IF K1>=A2 THEN K3=1,04E-8
1560    IF ABS(K1)>.5 THEN 1580
1565    K4=K1*K1
1570    K3=1-K1*((P9(1)*K4+P9(2))*K4+P9(3))/((K4+Q9(1))*K4+Q9(2))
1575    RETURN
1580    K4=ABS(K1)
1590    K3=((((R9(1)*K4+R9(2))*K4+R9(3))*K4+R9(4))*K4+R9(5))/((((K4+S9
(1))*K4+S9(2))*K4+S9(3))*K4+S9(4))*EXP(-K1*K1)
```

```
1600    RETURN
1605  ! K6=1/(K1*K1).  NOT USED PRESENTLY--FOR USE WHEN K1>4.
1610  ! K3=(B1+K6*(((V9(1)*K6+V9(2))*K6+V9(3))*K6+V9(4))/(((K6+W9(1))*
K6+W9(2))*K6+W9(3)))*EXP(-K1*K1)/K4
1620  X7=0  ! CRUTCHER TABLE CHECK
1625  INPUT R,H,K,S1,S2
1630  GOSUB 100
1635  PRINT R;H;K;S1;S2
1640  R4=R
1645  X7=X7+.1
1650  R=R4*X7
1655  GOSUB 690
1660  IMAGE DD.DD,2X,D.DDDDDD
1665  PRINT USING 1660;X7;P
1670  IF P<=.999999 THEN 1645
1675  BEEP
1680  END
```

ELLCV3 – HP 9845

```
100    ! THIS PROGRAM IS CALLED "ELLCV3". IT SUPPLIES
         THE ELLIPTICAL COVERAGE FUNCTION:P(R,H,K,S1,S2).
105    ! P DENOTES THE PROBABILITY OF A SHOT, NORMALLY
        DISTRIBUTED WITH MEAN (0,0) AND STANDARD
110    ! DEVIATIONS S1,S2 IN THE X AND Y DIRECTIONS,
        RESPECTIVELY, FALLING IN A CIRCLE IN THE
115    ! XY-PLANE OF RADIUS R AND CENTERED AT (H,K). THE
        INPUT IS R,H,K,S1,S2. THE OUTPUT IS P.
120    ! PROGRAM IS SET FOR 3-DECIMAL-DIGIT ACCURACY IN P.
125    ! ELLCV3 USES ERFC WITH 9 DIGIT RELATIVE ACCURACY.
130    ! SOURCES: NWL REPORT #1710, AUG.1960. MATH OF
        COMP. OCT. 1961, PP. 375,382.
135    ! INPUT R,H,K,S1,S2
140    ! "ELLCV3" CONSTRUCTED IN COLLABORATION WITH ALFRED MORRIS.
145    OPTION BASE 1
150    DIM P9(3),Q9(2),R9(5),S9(4),X(21),Y(21)
155    P9(1)=.316652890658
160    P9(2)=1.72227577039
165    P9(3)=21.3853322378
170    Q9(1)=7.8437457083
175    Q9(2)=18.9522572415
180    R9(1)=4.3187787405E-5
185    R9(2)=.56316961891
190    R9(3)=3.0317993362
195    R9(4)=6.8650184849
200    R9(5)=7.3738883116
205    S9(1)=5.3542167949
210    S9(2)=12.795529509
215    S9(3)=15.18490819
220    S9(4)=7.3739608908
225    X(1)=.238619186083
230    X(2)=.661209386466
235    X(3)=.932469514203
240    X(4)=.183434642496
245    X(5)=.525532409916
250    X(6)=.796666477414
255    X(7)=.960289856498
260    X(8)=.125233408511
265    X(9)=.367831498998
270    X(10)=.587317954287
275    X(11)=.769902674194
280    X(12)=.90411725637
285    X(13)=.981560634247
290    X(14)=9.50125098376E-2
295    X(15)=.281603550779
300    X(16)=.458016777657
305    X(17)=.617876244403
310    X(18)=.755404408355
315    X(19)=.865631202388
320    X(20)=.944575023073
325    X(21)=.989400934992
330    Y(1)=.467913934573
335    Y(2)=.360761573048
340    Y(3)=.171324492379
```

```
345   Y(4)=.362683783378
350   Y(5)=.313706645878
355   Y(6)=.222381034453
360   Y(7)=.10122853629
365   Y(8)=.249147045813
370   Y(9)=.233492536538
375   Y(10)=.203167426723
380   Y(11)=.160078328543
385   Y(12)=.106939325995
390   Y(13)=4.71753363865E-2
395   Y(14)=.189450610455
400   Y(15)=.182603415045
405   Y(16)=.169156519395
410   Y(17)=.149595988817
415   Y(18)=.124628971256
420   Y(19)=9.51585116825E-2
425   Y(20)=6.22535239386E-2
430   Y(21)=2.71524594118E-2
435   A=3.291
440   A1=3.89895
445   A2=2.898
450   A3=3.7
455   B1=.564189583548    ! 1/SQR(PI)
460   B=1.41421356237     ! SQR(2)
465   B2=15.2018111       !B2=A1*A1
470   P=0
475   Z3=.001*S1*S2
480   IF R*R<=Z3 THEN RETURN
485   H2=H*H+K*K
490   D=MAX(S1,S2)
495   T=R-A1*D
500    ! PROCEED TO SEE IF P=0 OR P=1
505   IF T<0 THEN 525
510   IF T*T<H2 THEN 525
515   P=1
520   RETURN
525   H8=ABS(H)
530   K8=ABS(K)
535   IF R-H8+A*S1<=0 THEN RETURN
540   IF R-K8+A*S2<=0 THEN RETURN
545   S0=SQR(H2)
550   IF S1<>S2 THEN 570
555   H8=S0
560   K8=0
565   IF R-H8+A*S1<=0 THEN RETURN
570   IF S0<=R THEN 690
575   D=(S0-R)/D
580   IF R*R*EXP(-.5*D*D)>Z3 THEN 690
585   RETURN
590   IF S0<R+A1*MIN(S1,S2) THEN 690
595   IF H8*K8=0 THEN 690
600   H9=H8/S1
605   K9=K8/S2
```

```
610   D=H9*H9+K9*K9
615   IF D<=B2 THEN 690
620   Z2=R/S2
625   Q=S2/S1
630   Q1=Q*Q
635   F=Q1*H9*H9+K9*K9
640   Z1=Z2*Z2*F/D
645   Z=D-Z1-B2
650   IF Z<0 THEN 660
655   IF Z*Z-4*Z1*B2>=0 THEN RETURN
660   T1=H8*H8+Q1*K8*K8
665   Z8=B2*S1*S1*T1/H2
670   R2=R*R
675   Y=H2-R2-Z8
680   IF Y<=0 THEN 690
685   IF Y*Y-4*R2*Z8>=0 THEN RETURN
690   Z8=0  !  FIND LIMITS OF INTEGRATION
695   Z=K8+A3*S2
700   H3=K8-A3*S2
705   S0=S1
710   S9=S2
715   Z=R-Z
720   D1=0
725   IF Z>0 THEN D1=SQR(Z/R)
730   IF H3>=0 THEN 750
735   H5=0
740   E3=1-D1
745   GOTO 760
750   E3=SQR(1-H3/R)-D1
755   H5=1
760   IF Z8<>0 THEN 800
765   Z8=1
770   F=E3
775   T=D1
780   Z=H8+A3*S1
785   H6=H5
790   H3=H8-A3*S1
795   GOTO 715
800   IF F>=E3 THEN 845
805   E3=F
810   D1=T
815   S9=S1
820   Z8=H8
825   S0=S2
830   H8=K8
835   K8=Z8
840   H5=H6
845   E3=.5*E3  ! BEGIN GAUSSIAN INTEGRATION
850   N=E3*R*(.34/S0+1/(.025*ABS(R-K8)+5*S9))
855   Z2=R/(B*S9)
860   R8=R/(B*S0)
865   H8=H8/(B*S0)
870   K8=K8/(B*S9)
```

```
875   IF N<2 THEN 895
880   J=13
885   N1=8
890   GOTO 945
895   IF N<.675 THEN 915
900   J=7
905   N1=6
910   GOTO 945
915   IF N<.5 THEN 935
920   J=3
925   N1=4
930   GOTO 945
935   J=0
940   N1=3
945   Z=Z3=0
950   Y=B1*E3*R8
955   K9=.000015
960   H9=1.99997
965   G3=0
970   M=N1+N1
975   I=-N1
980   IF K8=0 THEN 1160
985   Z3=0
990   FOR L=1 TO M
995     IF I=0 THEN I=1
1000    T=E3*(SGN(I)*X(J+ABS(I))+1)+D1
1005    T9=T*T
1010    T1=R8*(1-T9)
1015    T2=H8-T1
1020    T4=EXP(-T2*T2)
1025    IF H8<>0 THEN 1040
1030    T4=T4+T4
1035    GOTO 1055
1040    IF H5<>0 THEN 1055
1045    T2=H8+T1
1050    T4=T4+EXP(-T2*T2)
1055    IF Z<>0 THEN 1085
1060    Z1=Z2*T*SQR(2-T9)
1065    K1=K6-Z1
1070    IF ABS(K1)<A2 THEN 1095
1075    IF K1>0 THEN 1100
1080    Z=1
1085    K5=H9
1090    GOTO 1105
1095    GOSUB 1305
1100    K5=K3
1105    K1=K8+Z1
1110    IF K1<A2 THEN 1125
1115    K5=K5-K9
1120    GOTO 1135
1125    GOSUB 1305
1130    K5=K5-K3
1135    G3=G3+K5*T4*T*Y(J+ABS(I))
```

```
1140     I=I+1
1145     NEXT L
1150  P=Y*G3
1155  RETURN
1160  FOR L=1 TO M
1165     IF I=0 THEN I=1
1170     T=E3*(SGN(I)*X(J+ABS(I))+1)+D1
1175     T9=T*T
1180     T1=R8*(1-T9)
1185     T2=H8-T1
1190     T4=EXP(-T2*T2)
1195     IF H8<>0 THEN 1210
1200     T4=T4+T4
1205     GOTO 1225
1210     IF H5<>0 THEN 1225
1215     T2=H8+T1
1220     T4=T4+EXP(-T2*T2)
1225     IF Z<>0 THEN 1245
1230     K1=Z2*T*SQR(2-T9)
1235     IF K1<A2 THEN 1255
1240     Z=1
1245     K5=H9
1250     GOTO 1265
1255     GOSUB 1305
1260     K5=2*(1-K3)
1265     G3=G3+K5*T4*T*Y(J+ABS(I))
1270     I=I+1
1275     NEXT L
1280     P=Y*G3
1285     RETURN
1290     REM  CODY FOR K3=ERFC(K1)--9 DIGITS
1295     ! IF K1<=-A2 THEN K3=H9.
1300     ! IF K1>=A2 THEN K3=K9.
1305     IF ABS(K1)>.5 THEN 1325
1310     K4=K1*K1
1315     K3=1-K1*((P9(1)*K4+P9(2))*K4+P9(3))/((K4+Q9(1))*K4+Q9(2))
1320     RETURN
1325     K4=ABS(K1)
1330     K3=((((R9(1)*K4+R9(2))*K4+R9(3))*K4+R9(4))*K4+R9(5))/((((K4+S9
(1))*K4+S9(2))*K4+S9(3))*K4+S9(4))*EXP(-K1*K1)
1335     IF K1<0 THEN K3=2-K3
1340     RETURN
1345  X7=0     ! CRUTCHER TABLE CHECK
1350  INPUT R,H,K,S1,S2
1355  GOSUB 100
1360  PRINT R;H;K;S1;S2
1365  R4=R
1370  X7=X7+.1
1375  R=R4*X7
1380  GOSUB 470
1385  IMAGE DD.DD,2X,D.DDDD
1390  PRINT USING 1385;X7;P
1395  IF P<=.99999 THEN 1370
1400  BEEP
1405  END
```

ELLCV – HP 85

```
100 ! THIS PROGRAM IS CALLED"ELL
CV".IT SUPPLIES THE ELLIPTIC
AL COVERAGE FUNCTION:P(R,H,K
,S1,S2)
105 ! P DENOTES THE PROBABILITY
OF A SHOT, NORMALLY DISTRIBU
TED WITH MEAN (0,0) AND STAN
DARD
110 ! DEVIATIONS S1,S2 IN THE X
AND Y DIRECTIONS, RESPECTIVE
LY, FALLING IN A CIRCLE IN T
HE
115 ! XY-PLANE OF RADIUS R AND C
ENTERED AT (H,K). THE INPUT
IS R,H,K,S1,S2. THE OUTPUT I
S P
120 ! PROGRAM IS SET FOR 6-DECIM
AL-DIGIT ACCURACY IN P.
125 ! ELLCV USES ERFC WITH 9-DIG
IT RELATIVE ACCURACY.
130 ! SOURCES: NWL REPORT # 1710
,AUG.1960. MATH OF COMP. OCT
. 1961, PP. 375,382.
135 ! "ELLCV" CONSTRUCTED IN COL
LABORATION WITH ALFRED H. MO
RRIS.
140 OPTION BASE 1
145 DIM P9(3),Q9(2),R9(5),S9(4),
X(43),Y(43)
150 P9(1)=.316652890658 @ P9(2)=
1.72227577039 @ P9(3)=21.385
3322378
155 Q9(1)=7.8437457083 @ Q9(2)=1
8.9522572415
160 R9(1)=4.3187787405E-5 @ R9(2
)=.56316961891 @ R9(3)=3.031
7993362
165 R9(4)=6.8650184849 @ R9(5)=7
.3738883116
170 S9(1)=5.3542167949 @ S9(2)=1
2.795529509
175 S9(3)=15.18490819 @ S9(4)=7.
3739608908
190 X(1)=.238619186083 @ X(2)=.6
61209386466 @ X(3)=.93246951
4203 @ X(4)=.183434642496
195 X(5)=.525532409916 @ X(6)=.7
96666477414 @ X(7)=.96028985
6498 @ X(8)=.125233408511
200 X(9)=.367831498998 @ X(10)=.
587317954287 @ X(11)=.769902
674194 @ X(12)=.90411725637
205 X(13)=.981560634247 @ X(14)=
9.50125098376E-2 @ X(15)=.28
1603550779 @ X(16)=.45801677
7657
210 X(17)=.617876244403 @ X(18)=
.755404408355 @ X(19)=.86563
1202388 @ X(20)=.94457502307
3
215 X(21)=.989400934992 @ X(22)=
7.65265211335E-2 @ X(23)=.22
7785851142 @ X(24)=.37370608
8715
220 X(25)=.510867001951 @ X(26)=
.636053680727 @ X(27)=.74633
190646 @ X(28)=.839116971822
225 X(29)=.912234428251 @ X(30)=
.963971927278 @ X(31)=.99312
8599185
230 X(32)=6.40568928626E-2 @ X(3
3)=.191118867474 @ X(34)=.31
5042679696 @ X(35)=.43379350
7626
235 X(36)=.545421471389 @ X(37)=
.648093651937 @ X(38)=.74012
4191579 @ X(39)=.82000198597
4
240 X(40)=.886415527004 @ X(41)=
.938274552003 @ X(42)=.97472
8555971 @ X(43)=.99518721999
7
245 Y(1)=.467913934573 @ Y(2)=.3
60761573048 @ Y(3)=.17132449
2379 @ Y(4)=.362683783378
250 Y(5)=.313706645878 @ Y(6)=.2
22381034453 @ Y(7)=.10122853
629
255 Y(8)=.249147045813 @ Y(9)=.2
33492536538 @ Y(10)=.2031674
26723 @ Y(11)=.160078328543
260 Y(12)=.106939325995 @ Y(13)=
4.71753363865E-2 @ Y(14)=.18
9450610455 @ Y(15)=.18260341
5045
265 Y(16)=.169156519395 @ Y(17)=
.149595988817 @ Y(18)=.12462
8971256 @ Y(19)=9.5158511682
5E-2
270 Y(20)=6.22535239386E-2 @ Y(2
1)=2.71524594118E-2 @ Y(22)=
.152753387131 @ Y(23)=.14917
2986473
```

ELLCV – HP 85

```
275 Y(24)=.142096109318 @ Y(25)=
    .131688638449 @ Y(26)=.11819
    4531962 @ Y(27)=.10193011981
    7
280 Y(28)=8.32767415767E-2 @ Y(2
    9)=6.26720483341E-2 @ Y(30)=
    4.06014298004E-2 @ Y(31)=1.7
    6140071392E-2
285 Y(32)=.127938195347 @ Y(33)=
    .125837456347 @ Y(34)=.12167
    0472928 @ Y(35)=.11550566805
    4
290 Y(36)=.107444270116 @ Y(37)=
    9.76186521041E-2 @ Y(38)=.08
    6190161532 @ Y(39)=7.3346481
    4111E-2
295 Y(40)=5.92985849154E-2 @ Y(4
    1)=4.42774388174E-2 @ Y(42)=
    2.85313886289E-2 @ Y(43)=.01
    23412298
300 A=4.892 @ A1=5.387 @ A2=3.87
    75 @ A3=5.16
305 B1=.564189583548 @ B=1.41421
    356237 @ B2=29.019769 @ ! B2
    =A1*A1
310 Z3=.000001*S1*S2 @ P=0
315 IF R*R<=Z3 THEN RETURN
320 H2=H*H+K*K
325 D=MAX(S1,S2)
330 T=R-A1*D
335 ! PROCEED TO SEE IF P=0 OR 1

340 IF T<0 THEN 355
345 IF T*T<H2 THEN 355
350 P=1 @ RETURN
355 H8=ABS(H) @ K8=ABS(K)
360 IF R-H8+A*S1<=0 THEN RETURN
365 IF R-K8+A*S2<=0 THEN RETURN
370 S0=SQR(H2) @ IF S1#S2 THEN 3
    80
375 H8=S0 @ K8=0 @ IF R+A*S1<=H8
     THEN RETURN
380 IF S0<=R THEN 435 ELSE D=(S0
    -R)/D
385 IF R*R*EXP(-(.5*D*D))<=Z3 TH
    EN RETURN
390 IF S0<R+A1*MIN(S1,S2) THEN 4
    35
395 IF H8*K8=0 THEN 435
400 H9=H8/S1 @ K9=K8/S2
405 D=H9*H9+K9*K9 @ IF D<=B2 THE
    N 435 ELSE Z2=R/S2
410 Q=S2/S1 @ Q1=Q*Q @ F=Q1*H9*H
    9+K9*K9 @ Z1=Z2*Z2*F/D
415 Z=D-Z1-B2 @ IF Z<0 THEN 425
420 IF Z*Z-4*Z1*B2>=0 THEN RETUR
    N
425 T1=H8*H8+Q1*K8*K8 @ Z8=B2*S1
    *S1*T1/H2 @ R2=R*R @ Y=H2-R2
    -Z8 @ IF Y<0 THEN 435
430 IF Y*Y-4*R2*Z8>=0 THEN RETUR
    N
435 Z8=0 ! FIND LIMITS OF INTEGR
    ATION.
440 Z=K8+A3*S2 @ H3=K8-A3*S2 @ S
    0=S1 @ S9=S2
445 Z=R-Z @ D1=0 @ IF Z>0 THEN D
    1=SQR(Z/R)
450 IF H3>=0 THEN 455 ELSE H5=0
    @ E3=1-D1 @ GOTO 460
455 E3=SQR(1-H3/R)-D1 @ H5=1
460 IF Z8#0 THEN 470
465 Z8=1 @ F=E3 @ T=D1 @ Z=H8+A3
    *S1 @ H6=H5 @ H3=H8-A3*S1 @
    GOTO 445
470 IF F>=E3 THEN 480 ELSE E3=F
475 D1=T @ S9=S1 @ Z8=H8 @ S0=S2
     @ H8=K8 @ K8=Z8 @ H5=H6
480 E3=.5*E3
485 ! GAUSSIAN INTEGRATION BEGIN
    S
490 N=E3*R*(.34/S0+1/(.025*ABS(R
    -K8)+5*S9))
495 Z2=R/(B*S9) @ R8=R/(B*S0) @
    H8=H8/(B*S0) @ K8=K8/(B*S9)
500 IF N<2.75 THEN 505 ELSE J=31
     @ N1=12 @ GOTO 530
505 IF N<1.35 THEN 510 ELSE J=21
     @ N1=10 @ GOTO 530
510 IF N<.75 THEN 515 ELSE J=13
    @ N1=8 @ GOTO 530
515 IF N<.35 THEN 520 ELSE J=7 @
     N1=6 @ GOTO 530
520 IF N<.15 THEN 525 ELSE J=3 @
     N1=4 @ GOTO 530
525 J=0 @ N1=3
530 Z=0 @ Z3=0 @ Y=B1*E3*R8
535 G3=0 @ H9=1.9999999792 @ K9=
    .0000000104 @ M=N1+N1 @ I=-N
    1
540 IF K8=0 THEN 635
545 FOR L=1 TO M
550 IF I=0 THEN I=1
555 T=E3*(SGN(I)*X(J+ABS(I))+1)+
    D1
560 T9=T*T @ T1=R8*(1-T9) @ T2=H
    8-T1 @ T4=EXP(-(T2*T2))
```

```
565 IF H8#0 THEN 570 ELSE T4=T4+
    T4 @ GOTO 580
570 IF H5#0 THEN 580
575 T2=H8+T1 @ T4=T4+EXP(-(T2*T2
    ))
580 IF Z#0 THEN 605
585 Z1=Z2*T*SQR(2-T9) @ K1=K8-Z1
590 IF ABS(K1)<A2 THEN 610
595 IF K1>0 THEN 625
600 Z=1
605 K5=H9 @ GOTO 615
610 K5=FNO(K1)
615 K1=K8+Z1 @ IF K1<A2 THEN K5=
    K5-FNO(K1) ELSE K5=K5-K9
620 G3=G3+K5*T4*T*Y(J+ABS(I))
625 I=I+1 @ NEXT L
630 P=Y*G3 @ RETURN
635 FOR L=1 TO M
640 IF I=0 THEN I=1
645 T=E3*(SGN(I)*X(J+ABS(I))+1)+
    D1
650 T9=T*T @ T1=R8*(1-T9) @ T2=H
    8-T1 @ T4=EXP(-(T2*T2))
655 IF H8#0 THEN 665
660 T4=T4+T4 @ GOTO 675
665 IF H5#0 THEN 675 ELSE T2=H8+
    T1
670 T4=T4+EXP(-(T2*T2))
675 IF Z#0 THEN 690
680 K1=Z2*T*SQR(2-T9) @ IF K1<A2
     THEN K5=2*(1-FNO(K1)) @ GOT
    O 695
685 Z=1
690 K5=H9
695 G3=G3+K5*T4*T*Y(J+ABS(I))
700 I=I+1 @ NEXT L
705 P=Y*G3
710 RETURN
715 REM CODY FOR FN0=K3=ERFC(K1)
720 ! IF K1<=-A2 THEN FNO=H9.
725 ! IF K1>=A2 THEN FNO=K9.
730 DEF FNO(K1)
735 IF ABS(K1)>.5 THEN 755
740 K4=K1*K1
745 FNO=1-K1*((P9(1)*K4+P9(2))*K
    4+P9(3))/((K4+Q9(1))*K4+Q9(2
    ))
750 GOTO 775
755 K4=ABS(K1)
760 K5=(((R9(1)*K4+R9(2))*K4+R9(
    3))*K4+R9(4))*K4+R9(5)
765 K5=K5*EXP(-(K1*K1))/((((K4+S
    9(1))*K4+S9(2))*K4+S9(3))*K4
    +S9(4))
770 IF K1<0 THEN FNO=2-K5 ELSE F
    NO=K5
775 FN END
780 X7=0 ! CRUTCHER TABLE CHECK
785 INPUT R,H,K,S1,S2
790 GOSUB 100
795 PRINT R;H;K;S1;S2
800 R4=R
805 X7=X7+.1
810 R=R4*X7
815 GOSUB 310
820 IMAGE DD.DD,2X,D.DDDDDD
825 PRINT USING 820 ; X7;P
830 IF P<=.999999 THEN 805 ELSE
    BEEP
835 END
```

ELLCV3 – HP 85

```
100 ! THIS PROGRAM IS CALLED "EL
    LCV3".IT SUPPLIES THE ELLIPT
    ICAL COVERAGE FUNCTION P(R,H
    ,K,S1,S2)
105 ! P DENOTES THE PROBABILITY
    OF A SHOT, NORMALLY DISTRIBU
    TED WITH MEAN (0,0) AND STAN
    DARD
110 ! DEVIATIONS S1,S2 IN THE X
    AND Y DIRECTIONS, RESPECTIVE
    LY,FALLING IN A CIRCLE IN TH
    E XY-PLANE
115 ! OF RADIUS R AND CENTERED A
    T (H,K). THE OUTPUT IS P.
120 ! THE PROGRAM IS SET FOR 3-D
    ECIMAL-DIGIT ACCURACY IN P.
125 ! ELLCV3 USES ERFC WITH 9-DI
    GIT RELATIVE ACCURACY.
130 ! SOURCES" NWL REPORT #1710,
     AUG.1960. MATH OF COMP. OCT
    .1961, PP. 375-382.
135 ! "ELLCV3" CONSTRUCTED IN CO
    LLABORATION WITH ALFRED H. M
    ORRIS
140 OPTION BASE 1
145 DIM P9(3),Q9(2),R9(5),S9(4),
    X(21),Y(21)
150 P9(1)=.316652890658 @ P9(2)=
    1.72227577039 @ P9(3)=21.385
    3322378
155 Q9(1)=7.8437457083 @ Q9(2)=1
    8.9522572415
160 R9(1)=4.3187787405E-5 @ R9(2
    )=.56316961891 @ R9(3)=3.031
    7993362
165 R9(4)=6.8650184849 @ R9(5)=7
    .3738883116
170 S9(1`=5.3542167949 @ S9(2)=1
    2.795529509 @ S9(3)=15.18490
    819 @ S9(4)=7.3739608908
175 X(1)=.238619186083 @ X(2)=.6
    61209386466 @ X(3)=.93246951
    4203 @ X(4)=.183434642496
180 X(5)=.525532409916 @ X(6)=.7
    96666477414 @ X(7)=.96028985
    6498 @ X(8)=.125233408511
185 X(9)=.367831498998 @ X(10)=.
    587317954287 @ X(11)=.769902
    674194 @ X(12)=.90411725637
190 X(13)=.981560634247 @ X(14)=
    9.50125098376E-2 @ X(15)=.28
    1603550779 @ X(16)=.45801677
    7657
195 X(17)=.617876244403 @ X(18)=
    .755404408355 @ X(19)=.86563
    1202388 @ X(20)=.94457502307
    3
200 X(21)=.989400934992
205 Y(1)=.467913934573 @ Y(2)=.3
    60761573048 @ Y(3)=.17132449
    2379 @ Y(4)=.362683783378
210 Y(5)=.313706645878 @ Y(6)=.2
    22381034453 @ Y(7)=.10122853
    629
215 Y(8)=.249147045813 @ Y(9)=.2
    33492536538 @ Y(10)=.2031674
    26723 @ Y(11)=.160078328543
220 Y(12)=.106939325995 @ Y(13)=
    4.71753363865E-2 @ Y(14)=.18
    9450610455 @ Y(15)=.18260341
    5045
225 Y(16)=.169156519395 @ Y(17)=
    .149595988817 @ Y(18)=.12462
    8971256 @ Y(19)=9.5158511682
    5E-2
230 Y(20)=6.22535239386E-2 @ Y(2
    1)=2.71524594118E-2
235 A=3.291 @ A1=3.89895 @ A2=2.
    898 @ A3=3.7
240 B1=.564189583548 @ B=1.41421
    356237 @ B2=15.2018111 @ ! B
    2=A1*A1
245 Z3=.001*S1*S2 @ P=0
250 IF R*R<=Z3 THEN RETURN
255 H2=H*H+K*K
260 D=MAX(S1,S2)
265 T=R-A1*D
270 ! PROCEED TO SEE IF P=0 OR 1
275 IF T<0 THEN 285
280 IF T*T>=H2 THEN P=1 @ RETURN
285 H8=ABS(H) @ K8=ABS(K)
290 IF R-H8+A*S1<=0 THEN RETURN
295 IF R-K8+A*S2<=0 THEN RETURN
300 S0=SQR(H2) @ IF S1#S2 THEN 3
    10
305 H8=S0 @ K8=0 @ IF R+A*S1<=H8
     THEN RETURN
310 IF S0<=R THEN 365 ELSE D=(S0
    -R)/D
315 IF R*R*EXP(-(.5*D*D))<=Z3 TH
    EN RETURN
320 IF S0<R+A1*MIN(S1,S2) THEN 3
    65
325 IF H8*K8=0 THEN 365
330 H9=H8/S1 @ K9=K8/S2
335 D=H9*H9+K9*K9 @ IF D<=B2 THE
    N 365 ELSE Z2=R/S2
```

```
340 Q=S2/S1 @ Q1=Q*Q @ F=Q1*H9*H
    9+K9*K9 @ Z1=Z2*Z2*F/D
345 Z=D-Z1-B2 @ IF Z<0 THEN 355
350 IF Z*Z-4*Z1*B2>=0 THEN RETUR
    N
355 T1=H8*H8+Q1*K8*K8 @ Z8=B2*S1
    *S1*T1/H2 @ R2=R*R @ Y=H2-R2
    -Z8 @ IF Y<0 THEN 365
360 IF Y*Y-4*R2*Z8>=0 THEN RETUR
    N
365 Z8=0 ! FIND LIMITS OF INTEGR
    ATION
370 Z=K8+A3*S2 @ H3=K8-A3*S2 @ S
    0=S1 @ S9=S2
375 Z=R-Z @ D1=0 @ IF Z>0 THEN D
    1=SQR(Z/R)
380 IF H3>=0 THEN 385 ELSE E3=1-
    D1 @ H5=0 @ GOTO 390
385 E3=SQR(1-H3/R)-D1 @ H5=1
390 IF Z8#0 THEN 400
395 Z8=1 @ F=E3 @ T=D1 @ Z=H8+A3
    *S1 @ H6=H5 @ H3=H8-A3*S1 @
    GOTO 375
400 IF F>=E3 THEN 410 ELSE E3=F
405 D1=T @ Z8=H8 @ S0=S2 @ S9=S1
     @ H8=K8 @ K8=Z8 @ H5=H6
410 E3=.5*E3
415 ! GAUSSIAN INTEGRATION BEGIN
    S
420 N=E3*R*(.34/S0+1/(.025*ABS(R
    -K8)+5*S9))
425 Z2=R/(B*S9) @ R8=R/(B*S0) @
    H8=H8/(B*S0) @ K8=K8/(B*S9)
430 IF N<2 THEN 435 ELSE J=13 @
    N1=8 @ GOTO 450
435 IF N<.675 THEN 440 ELSE J=7
    @ N1=6 @ GOTO 450
440 IF N<.5 THEN 445 ELSE J=3 @
    N1=4 @ GOTO 450
445 J=0 @ N1=3
450 Z=0 @ Z3=0 @ Y=B1*E3*R8
455 K9=.000015 @ H9=1.99997
460 G3=0 @ M=N1+N1 @ I=-N1
465 IF K8=0 THEN 560
470 FOR L=1 TO M
475 IF I=0 THEN I=1
480 T=E3*(SGN(I)*X(J+ABS(I))+1)+
    D1
485 T9=T*T @ T1=R8*(1-T9) @ T2=H
    8-T1 @ T4=EXP(-(T2*T2))
490 IF H8#0 THEN 495 ELSE T4=T4+
    T4 @ GOTO 505
495 IF H5#0 THEN 505
500 T2=H8+T1 @ T4=T4+EXP(-(T2*T2
    ))
505 IF Z#0 THEN 530
510 Z1=Z2*T*SQR(2-T9) @ K1=K8-Z1
515 IF K1<A2 THEN 535
520 IF K1>0 THEN 550
525 Z=1
530 K5=H9 @ GOTO 540
535 K5=FNA(K1)
540 K1=K8+Z1 @ IF K1<A2 THEN K5=
    K5-FNA(K1) ELSE K5=K5-K9
545 G3=G3+K5*T4*T*Y(J+ABS(I))
550 I=I+1 @ NEXT L
555 P=Y*G3 @ RETURN
560 FOR L=1 TO M
565 IF I=0 THEN I=1
570 T=E3*(SGN(I)*X(J+ABS(I))+1)+
    D1
575 T9=T*T @ T1=R8*(1-T9) @ T2=H
    8-T1 @ T4=EXP(-(T2*T2))
580 IF H8#0 THEN 590
585 T4=T4+T4 @ GOTO 600
590 IF H5#0 THEN 600 ELSE T2=H8+
    T1
595 T4=T4+EXP(-(T2*T2))
600 IF Z#0 THEN 615
605 K1=Z2*T*SQR(2-T9) @ IF K1<A2
     THEN K5=2*(1-FNA(K1)) @ GOT
    O 620
610 Z=1
615 K5=H9
620 G3=G3+K5*T4*T*Y(J+ABS(I))
625 I=I+1 @ NEXT L
630 P=Y*G3 @ RETURN
635 REM CODY FOR FNA=K3=ERFC(K1)
    -9-DIGITS.
640 ! IF K1<=-A2THEN FNA=H9. IF
    K1>A2 THEN FNA=K9.
645 DEF FNA(K1)
650 K5=0 @ IF ABS(K1)>.5 THEN 67
    0
655 K4=K1*K1
660 FNA=1-K1*((P9(1)*K4+P9(2))*K
    4+P9(3))/((K4+Q9(1))*K4+Q9(2
    ))
665 GOTO 690
670 K4=ABS(K1)
675 K5=(((R9(1)*K4+R9(2))*K4+R9(
    3))*K4+R9(4))*K4+R9(5)
680 K5=K5/((((K4+S9(1))*K4+S9(2)
    )*K4+S9(3))*K4+S9(4))*EXP(-(
    K1*K1))
685 IF K1<0 THEN FNA=2-K5 ELSE F
    NA=K5
690 FN END
```

## POLYCV

*Input:* (Data Statement for HP-9845) P8, P9, $M_x$, $M_y$, C, $S_x$, $S_y$, N
(Data Statement for HP-85) P8, P9, M1, M2, C, S1, S2, N

($M_x$, $M_y$) or (M1, M2) ≡ mean of the normal distribution

c ≡ correlation coefficient (contained in C)

$S_x$, $S_y$ or S1, S2 ≡ standard deviations $\sigma_x$ and $\sigma_y$

N = 1, K = 3 ➡ probability desired over a single angular region A1. Vertex of A1 is always given by $(x_1, y_1)$. Points $(x_2, y_2)$, $(x_3, y_3)$ are given in counterclockwise order about the vertex. POLYCV sets K = 3.

N = K ⩾ 3 ➡ probability desired over a polygon E. Vertices are specified in counterclockwise order. POLYCV sets K = N.

P8 is used to specify how the coordinates of the points defining E or A1 are stored. If P8 = 0, then $x_1, y_1, \ldots, x_K, y_K$ are stored consecutively in data statements immediately following the initial data statement above. POLYCV then stores $x_j$ in array element X(J) and $y_j$ in array element Y(J). If P8 ≠ 0, then it is assumed by POLYCV that the points are already stored in arrays X(*), Y(*). This is useful if the points are machine generated.

P9 is used to specify what output is desired as indicated below.

*Output:* P, A, W, I1 are always given as part of the output if N ⩾ 3. For N = 1, P and I1 are given.

P ≡ probability from a normal correlated bivariate distribution over an arbitrary polygon E or an angular region A1.

A ≡ area of E. If N = 1, then A is set to zero.

W ≡ winding number of E. For simple polygons, W = 1. If N = 1, W is set to zero. W is contained in W1.

I1 ≡ Error and Information Parameter:

I1 = 0, output acceptable.

I1 = −1, angular region A1, N = 1, may not be well-defined. The angular measure of A1 is close to 0 or $2\pi$. A result for P is given.

I1 = 1, angular region A1, N = 1, is not well-defined, i.e., at least one of the two points $(x_2, y_2)$, $(x_3, y_3)$ is too close to $(x_1, y_1)$. Input unacceptable. P is set to −5.

I1 = 2, two consecutive segments of E overlap. Output is O.K.

I1 = 3, the correlation coefficient, c, contained in C, does not satisfy $|c| < 1$. Input is unacceptable. P is set to −5.

P9 = 0, no additional output besides the above is given.

P9 = 1, the x, y coordinates of the vertices of E or the points of A1 are listed.

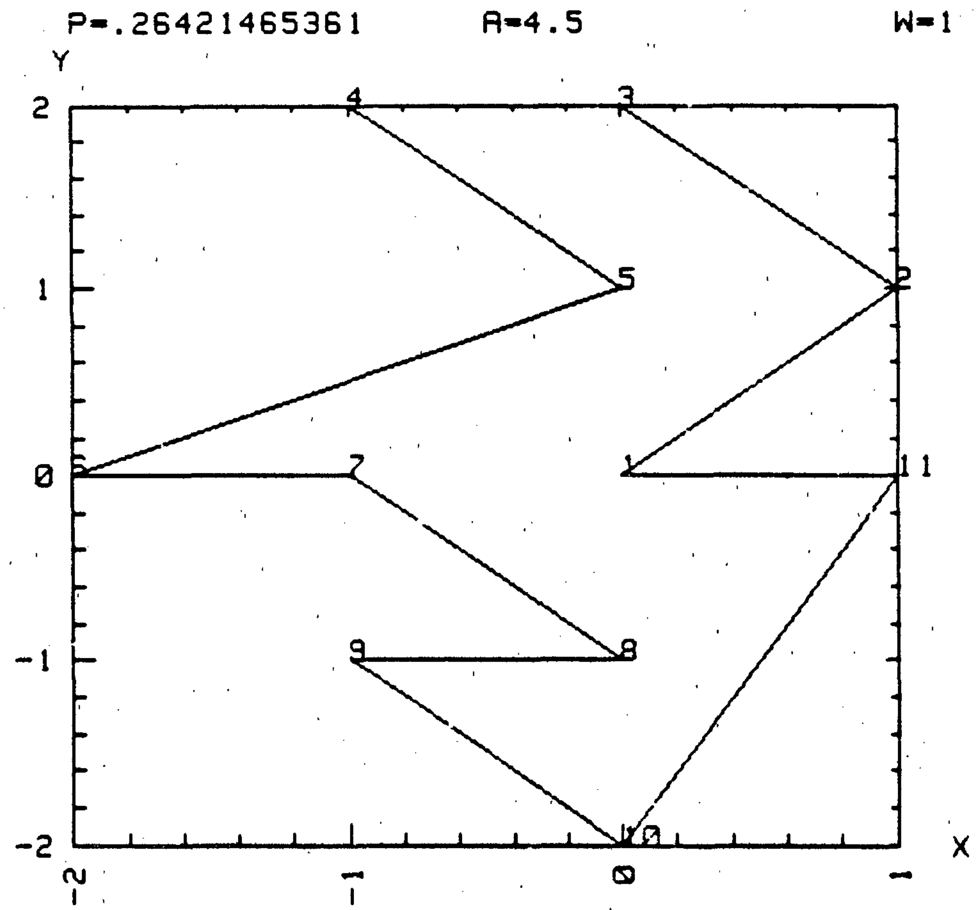
P9 < 0, a plot of E or A1 is given.

P9 > 1, both a listing of the x, y coordinates and a plot of E or A1 are given.

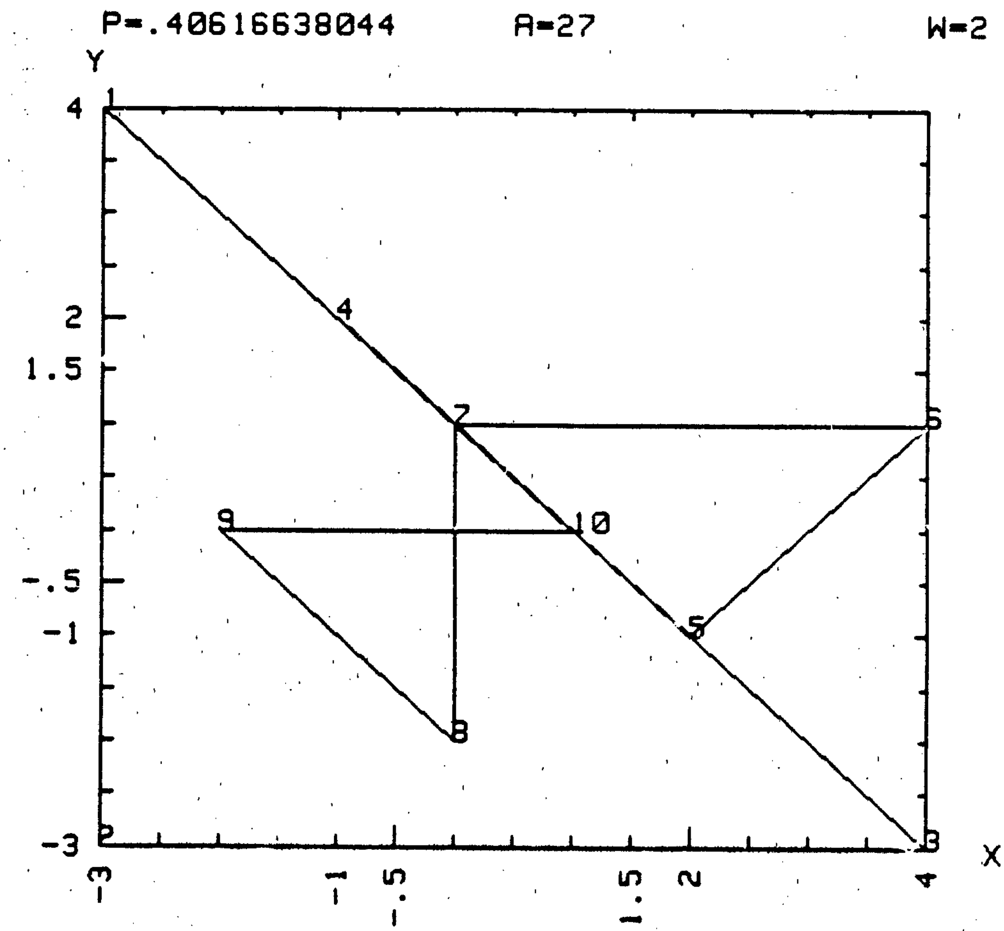
*Accuracy:* P given correctly to approximately 9-decimal digits.

## POLYCV HP-9845

```
P8= 0 P9= 2 Mx=-.5 My= 0 C= 0 Sx= 1 Sy= 2 N= 11

X( 1 )= 0              Y( 1 )= 0
X( 2 )= 1              Y( 2 )= 1
X( 3 )= 0              Y( 3 )= 2
X( 4 )=-1              Y( 4 )= 2
X( 5 )= 0              Y( 5 )= 1
X( 6 )=-2              Y( 6 )= 0
X( 7 )=-1              Y( 7 )= 0
X( 8 )= 0              Y( 8 )=-1
X( 9 )=-1              Y( 9 )=-1
X( 10 )= 0             Y( 10 )=-2
X( 11 )= 1             Y( 11 )= 0

P= .26421465361 A= 4.5 W= 1 I1= 0
```

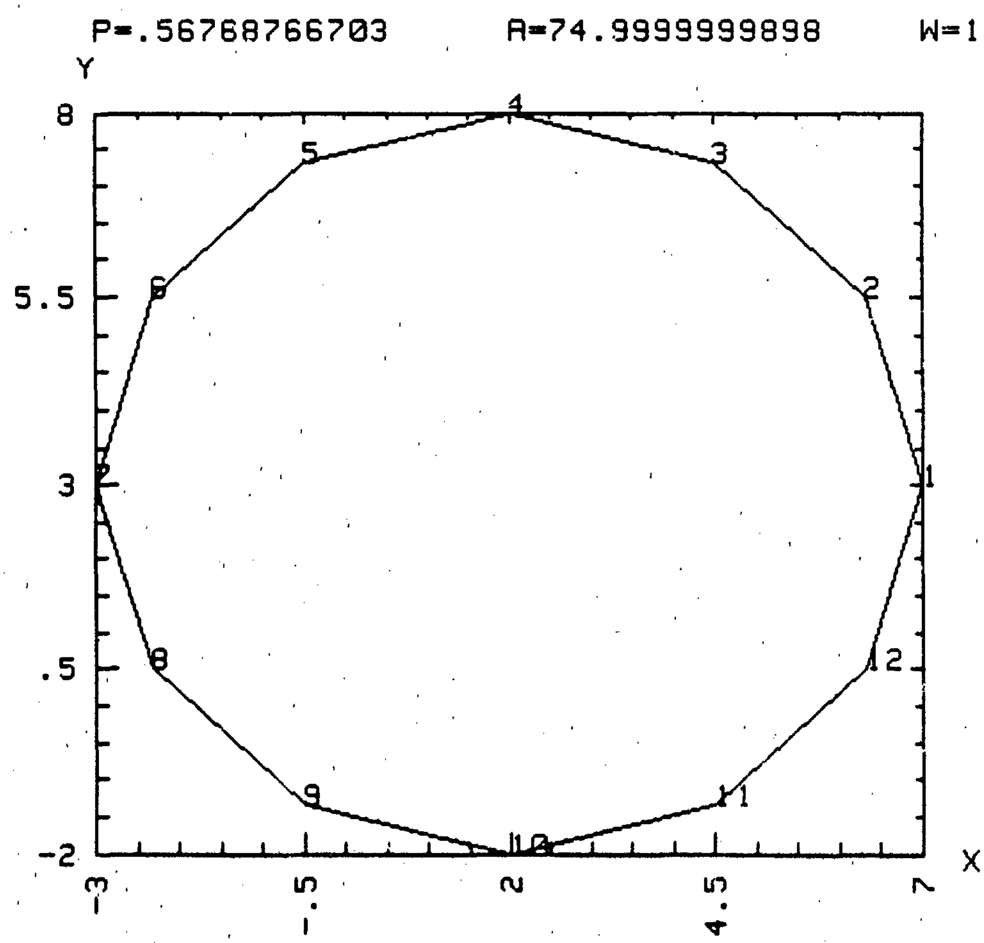


```
P8= 0 P9= 2 Mx= .5 My= 1 C= 0 Sx= 1 Sy= 3 N= 10

X( 1 )=-3              Y( 1 )= 4
X( 2 )=-3              Y( 2 )=-3
X( 3 )= 4              Y( 3 )=-3
X( 4 )=-1              Y( 4 )= 2
X( 5 )= 2              Y( 5 )=-1
X( 6 )= 4              Y( 6 )= 1
X( 7 )= 0              Y( 7 )= 1
X( 8 )= 0              Y( 8 )=-2
X( 9 )=-2              Y( 9 )= 0
X( 10 )= 1             Y( 10 )= 0

P= .40616638044 A= 27 W= 2 I1= 2
```

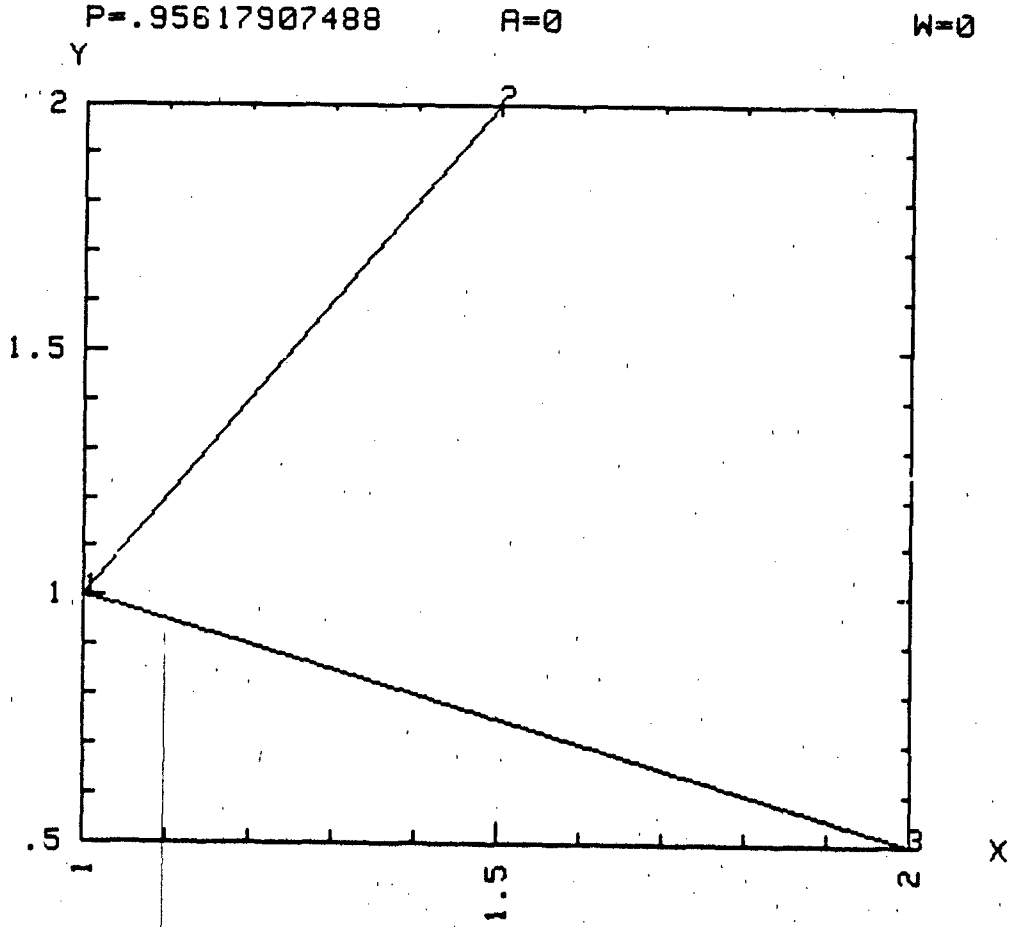


**POLYCV – HP 9845**

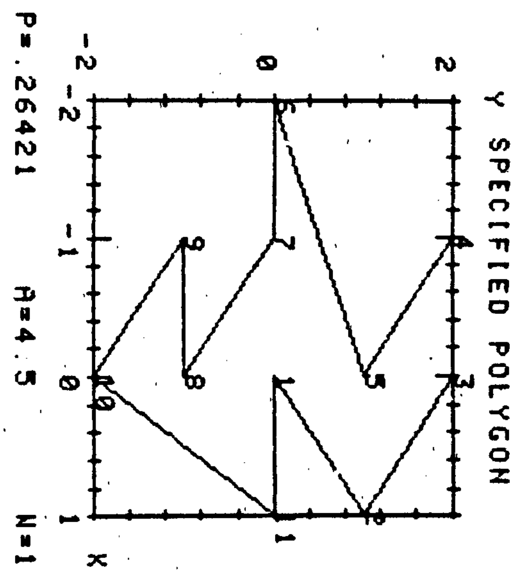
```
P8= 1 P9= 2 Mx= 0 My= 0 C= 0 Sx= 3 Sy= 2 N= 12

X( 1 )= 7                   Y( 1 )= 3
X( 2 )= 6.33012701885       Y( 2 )= 5.5
X( 3 )= 4.49999999987       Y( 3 )= 7.33012701885
X( 4 )= 1.99999999988       Y( 4 )= 7.99999999981
X( 5 )=-.50000000001        Y( 5 )= 7.33012701861
X( 6 )=-2.33012701873       Y( 6 )= 5.49999999966
X( 7 )=-2.9999999996        Y( 7 )= 2.99999999976
X( 8 )=-2.33012701838       Y( 8 )= .49999999999
X( 9 )=-.49999999946        Y( 9 )=-1.33012701861
X( 10 )= 2.00000000035      Y( 10 )=-1.99999999939
X( 11 )= 4.5                Y( 11 )=-1.33012701815
X( 12 )= 6.3301270185       Y( 12 )= .50000000074

P= .56768766703 A= 74.9999999898 W= 1 I1= 0
```



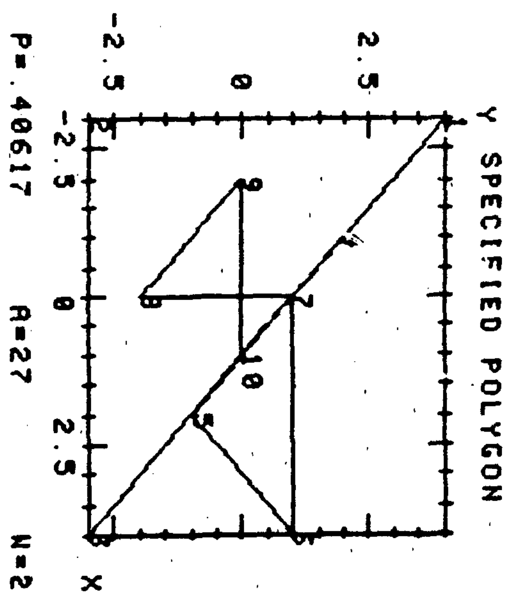
```
P8= 0 P9= 2 Mx= 0 My= 0 C= .5 Sx= 1 Sy= 2 N= 1

X( 1 )= 1                  Y( 1 )= 1
X( 2 )= 1.5                Y( 2 )= 2
X( 3 )= 2                  Y( 3 )= .5

P= .95617907488 A= 0 W= 0 I1= 0
```



```
INPUT: 0  2 -.5  0  0  1  2  11

X( 1 )= 0        Y( 1 )= 0
X( 2 )= 1        Y( 2 )= 1
X( 3 )= 0        Y( 3 )= 2
X( 4 )=-1        Y( 4 )= 2
X( 5 )= 0        Y( 5 )= 1
X( 6 )=-2        Y( 6 )= 0
X( 7 )=-1        Y( 7 )= 0
X( 8 )= 0        Y( 8 )=-1
X( 9 )=-1        Y( 9 )=-1
X( 10 )= 0       Y( 10 )=-2
X( 11 )= 1       Y( 11 )= 0

 .264214653623  4.5  1  0
```

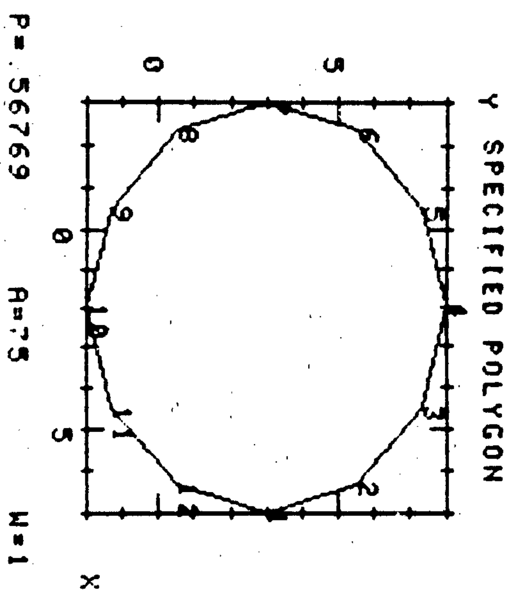


```
INPUT: 0  2  .5  1  0  1  3  10
X( 1 )=-3          Y( 1 )= 4
X( 2 )=-3          Y( 2 )=-3
X( 3 )= 4          Y( 3 )=-3
X( 4 )=-1          Y( 4 )= 2
X( 5 )= 2          Y( 5 )=-1
X( 6 )= 4          Y( 6 )= 1
X( 7 )= 0          Y( 7 )= 1
X( 8 )= 0          Y( 8 )=-2
X( 9 )=-2          Y( 9 )= 0
X( 10 )= 1         Y( 10 )= 0
 .40616638001   27   2   2
```

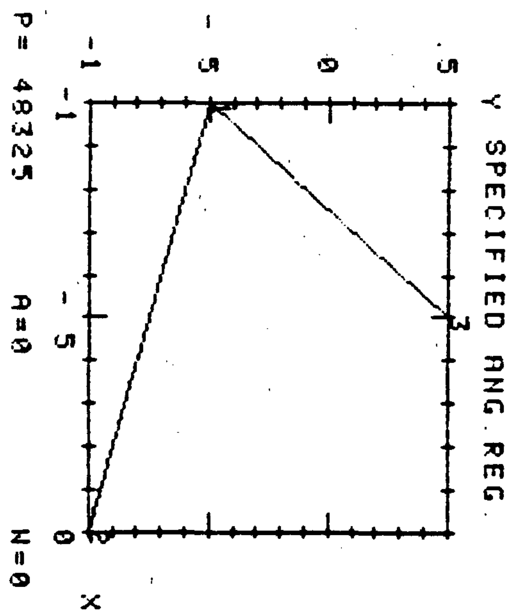


**POLYCV – HP 85**

```
INPUT: 1  2  0  0  0  3  2  12
X( 1 )= 7         Y( 1 )= 3
X( 2 )= 6.33012701893
                  Y( 2 )= 5.5
X( 3 )= 4.50000000001
                  Y( 3 )=
 7.33012701893
X( 4 )= 2         Y( 4 )= 8
X( 5 )=-.5        Y( 5 )=
 7.33012701893
X( 6 )=-2.33012701893
                  Y( 6 )=
 5.50000000001
X( 7 )=-3         Y( 7 )= 3
X( 8 )=-2.33012701893
                  Y( 8 )= .5
X( 9 )=-.50000000001
                  Y( 9 )=
-1.33012701893
X( 10 )= 2        Y( 10 )=-2
X( 11 )= 4.5      Y( 11 )=
-1.33012701893
X( 12 )= 6.33012701893
                  Y( 12 )=
 .49999999999

 .567687666698  75  1  0
```



```
INPUT: 0  2  0  0   5  1  2  1

X( 1 )=-1         Y( 1 )=-.5
X( 2 )= 0         Y( 2 )=-1
X( 3 )=-.5        Y( 3 )= .5

  483248792488  0
```



**POLYCV – HP 9845**

```
100   ! THIS PROGRAM IS CALLED "POLYCV". IT SUPPLIES
        THE PROBABILITY P OF A SINGLE SHOT, NORMALLY
105   ! DISTRIBUTED IN THE XY-PLANE WITH MEAN (Mx,My)
        STANDARD DEVIATIONS Sx,Sy AND CORRELATION
110   ! COEFFICIENT C, FALLING IN AN ARBITRARY POLYGON
        E OR IN A SEMI-INFINITE ANGULAR REGION A1 IN THE
115   ! PLANE. E IS SPECIFIED BY K POINTS (X(J),Y(J)). FOR J=
        1 TO K. A1 IS SPECIFIED BY 3 POINTS (X(J),Y(J)) WITH
120   ! N=1 AND THE VERTEX OF A1 GIVEN BY (X(1),Y(1)). THE
        POINTS MUST BE GIVEN IN COUNTERCLOCKWISE ORDER.
125   ! THE INITIAL INPUT IS:P8,P9,Mx,My,C,Sx,Sy,N. IT IS STORED
        AS DATA BEGINNING AT 2000. IF N=K>=3
130   ! THEN THE INPUT SPECIFIES A POLYGON E. IF N=1 (WITH
        K=3), THEN AN ANGULAR REGION A1 IS GIVEN.
135   ! IF P8=0, THEN THE COORDINATES X(J),Y(J), FOR J=1 TO K,
        ARE STORED IN DATA STATEMENTS IMMEDIATELY FOLLOWING
140   ! THE INITIAL INPUT DATA STATEMENT. POLYCV STORES THEM IN
        ARRAYS X(*),Y(*). IF P8#0, THEN IT IS ASSUMED THE VERTEX
145   ! COORDINATES ARE ALREADY STORED IN THE ARRAYS X(*),Y(*).
        THIS IS USEFUL IF THE VERTEX COORDINATES ARE MACHINE
150   ! GENERATED. P IS GIVEN TO NINE DECIMAL-DIGIT-ACCURACY.
155   ! IF P9>=1 THEN LIST X(J),Y(J). IF P9>1 OR <0 THEN PLOT E
        OR A1. IF P9=0 THEN NO PLOT OR LIST.
160   ! THE OUTPUT IS: P,A,W,I1, WHERE A CONTAINS THE AREA
        OF E, W1 CONTAINS THE WINDING NUMBER W OF E, AND I1
165   ! IS AN ERROR INDICATOR. IF I1=0 OR 2 THEN THE OUTPUT IS
        ACCEPTABLE. IF I1 =1 OR -1, THE ANGULAR REGION A1 MAY NOT
170   ! BE WELL-DEFINED. IF I1=1 THE VERTEX AND ONE OF THE OTHER TWO
        POINTS MAY BE TOO CLOSE TO EACH OTHER. IF I1=-1, THEN THE
175   ! MEASURE OF A1 IS CLOSE TO 0 OR 2PI.
180   ! IF I1=2 THEN TWO CONSECUTIVE SIDES OF E OVERLAP. OUTPUT IS O.K.
        IF I1=3 THEN C*C>=1. IF I1=1 OR 3 THEN P SET TO -5.
185   ! SOURCES: NSWC/DL REPORT #3886,SEPT.1978. NSWC/DL REPORT#80-166,
        JUNE,1980. SIAM JN.SCI.STAT.COMPUT.,JUNE 1980,PP.179,186.
190   ! SIAM JN.SCI.STAT.COMPUT.,DEC,1982, PP. ?.
195   ! X(J) AND Y(J) DIMENSIONED AT 90. IF MORE POINTS ARE
        NEEDED TO SPECIFY E THEN MAKE CHANGES AT LINE 200.
200   DIM X(90),Y(90),U1(15)
205   U1(1)=4.08335517232E-7
210   U1(2)=-9.7186486416E-6
215   U1(3)=1.05787574481E-4
220   U1(4)=-7.04260243309E-4
225   U1(5)=3.24944543171E-3
230   U1(6)=-1.12323532148E-2
235   U1(7)=3.09199295521E-2
240   U1(8)=-7.149098378E-2
245   U1(9)=.145060043403
250   U1(10)=-.265638206366
255   U1(11)=.442851899329
260   U1(12)=-.666626670511
265   U1(13)=.886223733187
270   U1(14)=-.999999999776
275   U1(15)=.886226924931
```

```
280   Z9=1.12837916709
285   A1=2.71E-19
290   A2=1.34E-6
295   A4=7.311E-4
300   R2=19.201924
305   R1=.564189583546 ! 1/SQR(PI)
310   R3=4.9E-27
315   T7=3.14159265359
320   R4=.159154943092
325   T9=1.14472988585
330   R6=.28209479177
335   T8=7E-12
340   A7=3.2625E-11
345   ! INPUT: P8,P9,Mx,My,C,Sx,Sy,N,X(J),Y(J) (J=1 TO K)
350   READ P8,P9,Mx,My,C,Sx,Sy,N
355   EXIT ALPHA
360   EXIT GRAPHICS
365   PRINT "P8=";P8;"P9=";P9;"Mx=";Mx;"My=";My;"C=";C;"Sx=";Sx;"Sy=";Sy;"N=";N
370   PRINT
375   C7=1-C*C
380   IF C7>0 THEN 405
385   I1=3
390   P=-5
395   PRINT "I1=";3;"P=";-5
400   RETURN
405   C7=SQR(C7)
410   K=3
415   IF N<>1 THEN K=N
420   FOR J=1 TO K
425     IF P8=0 THEN READ X(J),Y(J)
430     IF P9>=1 THEN PRINT "X(";J;")=";X(J);TAB(25);"Y(";J;")=";Y(J)
435   NEXT J
440   IF (P9>1) OR (P9<0) THEN 1560
445   FOR J=1 TO K
450     Y(J)=(Y(J)-My)/Sy
455     X(J)=((X(J)-Mx)/Sx-C*Y(J))/C7
460   NEXT J
465   X(K+1)=X(1)
470   Y(K+1)=Y(1)
475   P=0
480   I1=0
485   A=0
490   K1=0
495   K=1
500   IF N<>1 THEN 550
505   X1=X(1)
510   Y1=Y(1)
515   X(3)=X(1)+X(1)-X(3)
520   Y(3)=Y(1)+Y(1)-Y(3)
525   U=X(2)-X(1)
530   V=Y(2)-Y(1)
535   W=X(1)-X(3)
540   Z=Y(1)-Y(3)
```

```
545   GOTO 590
550   Y1=Y(N+1)
555   X1=X(N+1)
560   U=X(2)-X(1)
565   V=Y(2)-Y(1)
570   X1=X(1)
575   Y1=Y(1)
580   W=X(1)-X(N)
585   Z=Y(1)-Y(N)
590   D1=W*W+Z*Z
595   IF D1>R3 THEN 620
600   IF N=1 THEN 1370
605   N=N-1
610   IF N=2 THEN 1175
615   GOTO 580
620   D2=U*U+V*V
625   IF D2>R3 THEN 665
630   IF N=1 THEN 1370
635   K=K+1
640   U=X(K+1)-X1
645   V=Y(K+1)-Y1
650   D2=U*U+V*V
655   IF D2<=R3 THEN 635
660   IF K=N-1 THEN 1175
665   A=X1*(Y(K+1)-Y(N))
670   B1=SQR(2*D1)
675   B2=SQR(2*D2)
680   P2=V*W-U*Z
685   C1=U*W+V*Z
690   GOSUB 1390
695   K1=K1+A5
700   L=0
705   B=.5*(X(K)*X(K)+Y(K)*Y(K))
710   IF B>A1 THEN 730
715   C2=0
720   P1=A5*R4-C2
725   GOTO 1145
730   G1=(W*X(K)+Z*Y(K))/B1
735   G2=(U*X(K)+V*Y(K))/B2
740   H1=(Z*X(K)-W*Y(K))/B1
745   H2=(V*X(K)-U*Y(K))/B2
750   IF ABS(P2)>2*B1*B2*A7 THEN 830
755   IF C1<0 THEN 780
760   IF ABS(A5)<=T8 THEN 770
765   IF G1<0 THEN 830
770   P1=0
775   GOTO 1145
780   I1=2
785   IF P2<0 THEN 810
790   H=H2
795   GOSUB 1435
800   P1=.5*E1
805   GOTO 1145
```

```
810   H=H1
815   GOSUB 1435
820   P1=-(.5*E1)
825   GOTO 1145
830   IF B<=A2 THEN 1050
835   IF G1<0 THEN 900
840   IF G2>=0 THEN 1090
845   G2=-G2
850   H2=-H2
855   IF ABS(H2)<=A4 THEN 880
860   H=-H2
865   GOSUB 1435
870   L=.5*E1
875   GOTO 1060
880   L=.5+R1*H2
885   GOTO 1060
890   L=.5-R1*H1
895   GOTO 1060
900   G1=-G1
905   H1=-H1
910   IF G2<0 THEN 940
915   IF ABS(H1)<=A4 THEN 890
920   H=H1
925   GOSUB 1435
930   L=.5*E1
935   GOTO 1060
940   G2=-G2
945   H2=-H2
950   IF ABS(H1)<=A4 THEN 1015
955   IF ABS(H2)<=A4 THEN 995
960   H=H1
965   GOSUB 1435
970   L=.5*E1
975   H=H2
980   GOSUB 1435
985   L=L-.5*E1
990   GOTO 1090
995   H=H1
1000  GOSUB 1435
1005  L=R1*H2-.5*(1-E1)
1010  GOTO 1090
1015  IF ABS(H2)<=A4 THEN 1040
1020  H=H2
1025  GOSUB 1435
1030  L=.5*(1-E1)-R1*H1
1035  GOTO 1090
1040  L=R1*(H2-H1)
1045  GOTO 1090
1050  C2=R6*(H2-H1)-R4*(G2*H2-G1*H1)
1055  GOTO 720
1060  P2=-P2
1065  IF P2<=0 THEN 1085
1070  L=L-1
```

```
1075  A5=T7+A5
1080  GOTO 1090
1085  A5=A5-T7
1090  IF B>=R2 THEN 1140
1095  C3=A5
1100  C4=.5*A5
1105  M=15
1110  F=0
1115  A6=H2-H1
1120  C5=A6
1125  GOTO 1310
1130  P1=L+EXP(-(B+T9))*(C4-S2)
1135  GOTO 1145
1140  P1=L
1145  IF K<>N THEN 1235
1150  IF N<>1 THEN 1190
1155  P=ABS(P1)
1160  IF K1>0 THEN P=ABS(1-P)
1165  C4=W1=0
1170  IF ABS(K1)<1E-11 THEN I1=-1
1175  PRINT
1180  PRINT "P=";P;"I1=";I1
1185  GOTO 1485
1190  P=P-P1
1195  K1=K1*R4
1200  A=.5*A
1205  IF K1<0 THEN 1220
1210  W1=INT(K1+.1)
1215  GOTO 1225
1220  W1=INT(K1+.9)
1225  P=P+W1
1230  GOTO 1470
1235  W=U
1240  Z=V
1245  B1=B2
1250  X1=X(K+1)
1255  Y1=Y(K+1)
1260  Y2=Y(K)
1265  K=K+1
1270  U=X(K+1)-X1
1275  V=Y(K+1)-Y1
1280  D2=U*U+V*V
1285  IF D2<=R3 THEN 1265
1290  B2=SQR(2*D2)
1295  P=P-P1
1300  A=A+X1*(Y(K+1)-Y2)
1305  GOTO 680
1310  S2=U1(M)*A6
1315  M=M-1
1320  H2=H2*G2
1325  H1=H1*G1
1330  T=H2-H1
1335  F=F+B
```

```
1340  C6=(F*C3+T)/(16-M)
1345  S2=S2+U1(M)*C6
1350  IF M=1 THEN 1130
1355  C3=C5
1360  C5=C6
1365  GOTO 1315
1370  P=-5
1375  I1=1
1380  PRINT "I1=";1;"P=";-5
1385  RETURN
1390  IF ABS(C1)<=ABS(P2) THEN 1415  !A5=ARCTANGENT(P2/C1), (-PI,PI].
1395  IF (C1<0) AND (P2=0) THEN 1425
1400  A5=ATN(P2/C1)
1405  IF C1<0 THEN A5=A5+SGN(P2)*T7
1410  RETURN
1415  A5=SGN(P2)*(.5*T7-ATN(C1/ABS(P2)))
1420  RETURN
1425  A5=T7
1430  RETURN
1435  E1=0  ! E1=ERFC(H)
1440  C4=ABS(H)
1445  IF C4>=4.4 THEN 1460
1450  E1=((((((((((U1(1)*C4+U1(2))*C4+U1(3))*C4+U1(4))*C4+U1(5))*C4+U1
(6))*C4+U1(7))*C4+U1(8))*C4+U1(9))*C4+U1(10))*C4+U1(11))*C4+U1(12)
1455  E1=(((E1*C4+U1(13))*C4+U1(14))*C4+U1(15))*Z9*EXP(-H*H)
1460  IF H<0 THEN E1=2-E1
1465  RETURN
1470  C4=Sx*Sy*C7*A
1475  PRINT
1480  PRINT "P=";P;"A=";C4;"W=";W1;"I1=";I1
1485  IF (P9>1) OR (P9<0) THEN 1905
1490  BEEP
1495  RETURN
1500  PLOTTER IS "GRAPHICS"
1505  GRAPHICS
1510  U=X(1)
1515  V=U
1520  W=Y(1)
1525  Z=W
1530  FOR I=1 TO K
1535    IF X(I)>U THEN U=X(I)
1540    IF X(I)<V THEN V=X(I)
1545    IF Y(I)>W THEN W=Y(I)
1550    IF Y(I)<Z THEN Z=Y(I)
1555  NEXT I
1560  GCLEAR
1565  P1=U-V
1570  IF P1<>0 THEN 1590
1575  P1=1
1580  V=V-P1
1585  U=U+P1
1590  Y1=LGT(P1)
1595  X1=INT(Y1)
```

```
1600  Y1=FRACT(Y1)
1605  Y1=10^Y1
1610  IF X1<0 THEN Y1=10*Y1
1615  X1=10^X1
1620  IF Y1>=2 THEN 1635
1625  T1=.1*X1
1630  GOTO 1670
1635  IF Y1>=4 THEN 1650
1640  T1=.2*X1
1645  GOTO 1670
1650  IF Y1>=5 THEN 1665
1655  T1=.4*X1
1660  GOTO 1670
1665  T1=.5*X1
1670  T3=INT(V/T1)*T1
1675  T4=U
1680  IF FRACT(T4/T1)<>0 THEN T4=(INT(T4/T1)+1)*T1
1685  P2=W-Z
1690  IF P2<>0 THEN 1710
1695  P2=1
1700  Z=Z-P2
1705  W=W+P2
1710  Y1=LGT(P2)
1715  X1=INT(Y1)
1720  Y1=FRACT(Y1)
1725  Y1=10^Y1
1730  IF X1<0 THEN Y1=10*Y1
1735  X1=10^X1
1740  IF Y1>=2 THEN 1755
1745  T2=.1*X1
1750  GOTO 1790
1755  IF Y1>=4 THEN 1770
1760  T2=.2*X1
1765  GOTO 1790
1770  IF Y1>=5 THEN 1785
1775  T2=.4*X1
1780  GOTO 1790
1785  T2=.5*X1
1790  T5=INT(Z/T2)*T2
1795  T6=0+W
1800  IF FRACT(T6/T2)<>0 THEN T6=(INT(T6/T2)+1)*T2
1805  SCALE T3-.35*P1,T4+.25*P1,T5-.25*P2,T6+.2*P2
1810  CLIP T3,T4,T5,T6
1815  LAXES T1,T2,T3,T5,5,5,4
1820  LAXES T1,T2,T4,T6,5,5,2
1825  IF N=1 THEN 1945
1830  MOVE X(1),Y(1)
1835  FOR I=1 TO N
1840    DRAW X(I),Y(I)
1845    LABEL VAL$(I)
1850    MOVE X(I),Y(I)
1855  NEXT I
1860  DRAW X(1),Y(1)
```

```
1865  MOVE U+T1,Z-.02*P1
1870  LABEL "X"
1875  MOVE T3-.02*P1,T6+.05*P2
1880  LABEL "Y"
1885  C8=T3
1890  C9=T6+.1*P2
1895  D3=P1
1900  GOTO 445
1905  MOVE C8,C9
1910  LABEL "P="&VAL$(P)
1915  MOVE C8+.5*D3,C9
1920  LABEL "A="&VAL$(C4)
1925  MOVE C8+D3,C9
1930  LABEL "W="&VAL$(W1)
1935  DUMP GRAPHICS
1940  RETURN
1945  MOVE X(3),Y(3)
1950  LABEL VAL$(3)
1955  MOVE X(3),Y(3)
1960  DRAW X(1),Y(1)
1965  LABEL VAL$(1)
1970  MOVE X(1),Y(1)
1975  DRAW X(2),Y(2)
1980  LABEL VAL$(2)
1985  GOTO 1865
```

**POLYCV – HP 85**

```
100 ! THIS PROGRAM IS CALLED "PO
    LYCV". IT SUPPLIES THE PROBA
    BILITY P OF A SINGLE SHOT, N
    ORMALLY
105 ! DISTRIBUTED IN THE XY-PLAN
    E WITH MEAN (M1,M2),STANDARD
     DEVIATIONS S1,S2 IN THE X A
    ND Y
110 ! DIRECTIONS,RESPECTIVELY,
    CORRELATION COEFFICIENT C,
115 ! FALLING IN AN ARBITRARY PO
    LYGON E OR A SEMI-INFINITE A
    NGULAR REGION A1 IN
120 ! THE XY-PLANE. E IS SPECIFI
    ED BY K POINTS (X(J),Y(J)),J
    =1 TO K. A1 IS SPECIFIED BY
    3 POINTS
125 ! (X(J),Y(J)) WITH THE VERTE
    X OF A1 GIVEN BY (X(1),Y(1))
     AND THE POINTS GIVEN IN COU
    NTER
130 ! CLOCKWISE ORDER. THE INITI
    AL INPUT IS:P8,P9,M1,M2,C,S1
    ,S2,N.
135 ! IT IS STORED AS DATA BEGIN
    NING AT 1085.
140 ! IF N=K>=3 THEN THE INPUT S
    PECIFIES E. IF N=1(WITH K=3)
    ,THEN A1 IS GIVEN.
145 ! IF P8=0, THEN THE COORDINA
    TES X(J),Y(J), FOR J=1 TO K,
     ARE STORED IN DATA STATEMEN
    TS IM-
150 ! MEDIATELY FOLLOWING THE IN
    ITIAL DATA STATEMENT. POLYCV
     STORES THEM IN ARRAYS X(*),
    Y(*).
155 ! IF P8#0, THEN IT IS ASSUME
    D THE VERTEX COORDINATES ARE
     ALREADY STORED IN ARRAYS X(
    *),Y(*).
160 ! THIS IS USEFUL IF THE VERT
    EX COORDINATES ARE MACHINE G
    ENERATED. IF P9>=1 THEN LIST
    X(J),Y(J).
165 ! IF P9>1 OR <0 THEN PLOT E
    OR A1.IF P9=0 THEN NO PLOT O
    R LIST.
170 ! THE OUTPUT IS:P,A,W,I1, WH
    ERE A CONTAINS THE AREA OF E
    , W1 CONTAINS THE WINDING NU
    MBER W
175 ! OF E, AND I1 IS AN ERROR I
    NDICATOR. IF I1=0 OR 2 THEN
    THE OUTPUT IS O.K. IF I1=1OR
    -1,THEN
180 ! ANGULAR REGION A1 IS NOT
    WELL-DEFINED. IF I1=1,THE VE
    RTEX OF A1 AND ONE OF THE OT
    HER TWO
185 ! POINTS ARE TOO CLOSE TO EA
    CH OTHER.IF I1=-1 THEN MEASU
    RE OF A1 IS CLOSE TO 0 OR 2P
    I.
190 ! IF I1=2 THEN 2 SIDES OF E
    OVERLAP.IF I1=3 THEN C*C>=1.
195 ! IF I1=1 OR 3 THEN P IS SET
    TO -5. IF OUTPUT IS O.K.,THE
    N P IS GIVEN TO 9-DECIMAL AC
    CURACY.
200 ! SOURCES: NSWC/DL REPORT #3
    886, SEPT.1978. NSWC/DL REPO
    RT#80-166, JUNE 1980. SIAM J
    N. SCI.
205 ! STAT. COMPUT.,JUNE 1980, P
    P.179,186. SIAM JN. SCI.STAT
    . COMPUT.,DEC. 1982, PP.?.
210 ! X(J) AND Y(J) ARE DIMENSIO
    NED AT 90. IF MORE POINTS AR
    E NEEDED TO SPECIFY E THEN M
    AKE
215 ! CHANGES AT LINE 225.
220 OPTION BASE 1
225 DIM X(90),Y(90),U1(15)
230 SHORT I9,J
235 A1=2.71E-19 @ A2=.00000134 @
     A4=.0007311 @ R2=19.201924
240 R1=.56418958355 @ R3=4.9E-27
     @ T7=PI @ R4=.159154943092
    @ T9=1.14472988585 @ R6=.282
    09479177
245 T8=.000000000007 @ A7=3.2625
    E-11 @ Z9=1.12837916709
250 U1(1)=.000000408336 @ U1(2)=
    -.000009718649 @ U1(3)=1.057
    875745E-4 @ U1(4)=-7.0426024
    33E-4
255 U1(5)=.003249445432 @ U1(6)=
    -.011232353215 @ U1(7)=.0309
    19929552 @ U1(8)=-.071490983
    78
260 U1(9)=.145060043403 @ U1(10)
    =-.265638206366 @ U1(11)=.44
    2851899329 @ U1(12)=-.666626
    670511
265 U1(13)=.886223733187 @ U1(14
    )=-.999999899776 @ U1(15)=.8
    36226924931
270 READ P8,P9,M1,M2,C,S1,S2,N
275 C7=1-C*C @ IF C7>0 THEN 285
    ELSE I1=3 @ P=-5
```

POLYCV–HP 85

```
280 PRINT "I1=";I1;"P=";P @ RETU
    RN
285 PRINT "INPUT:";P8;P9;M1;M2;C
    ;S1;S2;N @ PRINT @ C7=SQR(C7
    )
290 IF N#1 THEN K=N ELSE K=3 @ W
    1=0
295 FOR J=1 TO K
300 IF P8=0 THEN READ X(J),Y(J)
305 IF P9>=1 THEN PRINT "X(";J;"
    )=";X(J);TAB(16);"Y(";J;")="
    ;Y(J)
310 NEXT J
315 PRINT @ IF P9>1 OR P9<0 THEN
     770
320 FOR J=1 TO K
325 Y(J)=(Y(J)-M2)/S2 @ X(J)=((X
    (J)-M1)/S1-C*Y(J))/C7
330 NEXT J
335 X(K+1)=X(1) @ Y(K+1)=Y(1)
340 P=0 @ I1=0 @ A=0 @ K1=0 @ K=
    1
345 IF N#1 THEN 360
350 X1=X(1) @ X(3)=X(1)+X(1)-X(3
    ) @ Y(3)=Y(1)+Y(1)-Y(3) @ U=
    X(2)-X(1) @ V=Y(2)-Y(1)
355 W=X(1)-X(3) @ Y1=Y(1) @ Z=Y(
    1)-Y(3) @ GOTO 380
360 Y1=Y(N+1) @ X1=X(N+1)
365 U=X(2)-X(1) @ V=Y(2)-Y(1)
370 X1=X(1) @ Y1=Y(1)
375 W=X(1)-X(N) @ Z=Y(1)-Y(N)
380 D1=W*W+Z*Z
385 IF D1>R3 THEN 400
390 IF N=1 THEN 730
395 N=N-1 @ IF N=2 THEN 665 ELSE
     375
400 D2=U*U+V*V
405 IF D2>R3 THEN 430
410 IF N=1 THEN 730
415 K=K+1 @ U=X(K+1)-X1 @ V=Y(K+
    1)-Y1 @ D2=U*U+V*V
420 IF D2<=R3 THEN 415
425 IF K=N-1 THEN 665
430 A=X1*(Y(K+1)-Y(N)) @ B1=SQR(
    2*D1) @ B2=SQR(2*D2)
435 P2=V*W-U*Z @ C1=U*W+V*Z @ A5
    =ATN2(P2,C1) @ K1=K1+A5 @ L=
    0 @ B=.5*(X(K)*X(K)+Y(K)*Y(K
    ))
440 IF B>A1 THEN 450 ELSE C2=0
445 P1=A5*R4-C2 @ GOTO 645
450 G1=(W*X(K)+Z*Y(K))/B1 @ G2=(
    U*X(K)+V*Y(K))/B2 @ H1=(Z*X(
    K)-W*Y(K))/B1 @ H2=(V*X(K)-U
    *Y(K))/B2
455 IF ABS(P2)>2*B1*B2*A7 THEN 5
    10
460 IF C1<A THEN 480
465 IF ABS(A5)<=T8 THEN 475
470 IF G1>=0 THEN 475 ELSE 510
475 P1=0 @ GOTO 645
480 I1=2
485 IF P2<0 THEN 500
490 H=H2 @ GOSUB 735
495 P1=.5*E1 @ GOTO 645
500 H=H1 @ GOSUB 735
505 P1=-(.5*E1) @ GOTO 645
510 IF B<=A2 THEN 605
515 IF G1<0 THEN 545
520 IF G2>=0 THEN 625
525 G2=-G2 @ H2=-H2 @ IF ABS(H2)
    <=A4 THEN 535
530 H=-H2 @ GOSUB 735 @ L=.5*E1
    @ GOTO 610
535 L=.5+R1*H2 @ GOTO 610
540 L=.5-R1*H1 @ GOTO 610
545 G1=-G1 @ H1=-H1
550 IF G2<0 THEN 565
555 IF ABS(H1)<=A4 THEN 540
560 H=H1 @ GOSUB 735 @ L=.5*E1 @
     GOTO 610
565 G2=-G2 @ H2=-H2
570 IF ABS(H1)<=A4 THEN 590
575 IF ABS(H2)<=A4 THEN 585
580 H=H1 @ GOSUB 735 @ L=.5*E1 @
     H=H2 @ GOSUB 735 @ L=L-.5*E
    1 @ GOTO 625
585 H=H1 @ GOSUB 735 @ L=R1*H2-.
    5*(1-E1) @ GOTO 625
590 IF ABS(H2)<=A4 THEN 600
595 H=H2 @ GOSUB 735 @ L=.5*(1-E
    1)-R1*H1 @ GOTO 625
600 L=R1*(H2-H1) @ GOTO 625
605 C2=T6*(H2-H1)-R4*(G2*H2-G1*H
    1) @ GOTO 445
610 P2=-P2 @ IF P2<=0 THEN 620
615 L=L-1 @ A5=T7+A5 @ GOTO 625
620 A5=A5-T7
625 IF B>=R2 THEN 640
630 C3=A5 @ C4=.5*A5 @ M=15 @ F=
    0 @ A6=H2-H1 @ C5=A6 @ GOTO
    715
635 P1=L+EXP(-(B+T9))*(C4-S3) @
    GOTO 645
640 P1=L
645 IF K#N THEN 695
650 IF N#1 THEN 675 ELSE H1=0
655 IF K1<=0 THEN P=ABS(P1) ELSE
     P=ABS(1-ABS(P1))
```

POLYCV – HP 85

```
660 IF ABS(K1)<.000000000005 THE
    N I1=-1
665 IF N#1 THEN PRINT P;H1;W1;I1
     ELSE PRINT P;I1
670 BEEP @ IF P9=0 OR P9=1 THEN
    RETURN ELSE I9=P @ J=H1 @ GO
    TO 1055
675 P=P-P1 @ K1=K1*R4 @ A=.5*A @
     IF K1<0 THEN 685
680 W1=IP(K1+.1) @ GOTO 690
685 W1=IP(K1-.1)
690 P=P+W1 @ H1=S1*S2*C7*A @ GOT
    O 665
695 W=U @ Z=V @ B1=B2 @ X1=X(K+1
    ) @ Y1=Y(K+1) @ Y2=Y(K)
700 K=K+1 @ U=X(K+1)-X1 @ V=Y(K+
    1)-Y1 @ D2=U*U+V*V
705 IF D2<=R3 THEN 700
710 B2=SQR(2*D2) @ P=P-P1 @ A=A+
    X1*(Y(K+1)-Y2) @ GOTO 435
715 S3=U1(M)*A6
720 M=M-1 @ H2=H2*G2 @ H1=H1*G1
    @ T=H2-H1 @ F=F+B @ C6=(F*C3
    +T)/(16-M) @ S3=S3+U1(M)*C6
725 IF M=1 THEN 635 ELSE C3=C5 @
     C5=C6 @ GOTO 720
730 P=-5 @ I1=1 @ GOTO 665
735 E1=0 @ C4=ABS(H) @ IF ABS(H)
    >=4.4 THEN 755 ! E1=ERFC(H)-
    -9-DECIMAL DIGIT740 FOR I9=1
     TO 15
740 E1=(((((((U1(1)*C4+U1(2))*C4
    +U1(3))*C4+U1(4))*C4+U1(5))*
    C4+U1(6))*C4+U1(7))*C4+U1(8)
    )*C4
745 E1=(((((((E1+U1(9))*C4+U1(10
    ))*C4+U1(11))*C4+U1(12))*C4+
    U1(13))*C4+U1(14))*C4+U1(15)
    )*Z9
750 E1=E1*EXP(-(H*H))
755 IF H<0 THEN E1=2-E1
760 RETURN
765 BEEP 84,100 @ IF P9=0 THEN R
    ETURN
770 U=X(1) @ V=U @ W=Y(1) @ Z=W
775 FOR I=1 TO K
780 IF X(I)>U THEN U=X(I)
785 IF X(I)<V THEN V=X(I)
790 IF Y(I)>W THEN W=Y(I)
795 IF Y(I)<Z THEN Z=Y(I)
800 NEXT I
805 GCLEAR @ P1=U-V @ IF P1#0 TH
    EN 810 ELSE P1=1 @ V=V-P1 @
    U=U+P1
810 Y1=LGT(P1) @ X1=INT(Y1) @ Y1
    =FP(Y1)
815 Y1=10^Y1 @ IF X1<0 THEN Y1=1
    0*Y1
820 X1=10^X1 @ IF Y1>=2 THEN 830
825 T1=.1*X1 @ GOTO 855
830 IF Y1>=4 THEN 840
835 T1=.2*X1 @ GOTO 855
840 IF Y1>=5 THEN 850
845 T1=.4*X1 @ GOTO 855
850 T1=.5*X1
855 T3=INT(V/T1)*T1
860 T4=U @ IF FP(T4/T1)#0 THEN T
    4=(INT(T4/T1)+1)*T1
865 P2=W-Z @ IF P2#0 THEN 870 EL
    SE P2=1 @ Z=Z-P2 @ W=W+P2
870 Y1=LGT(P2) @ X1=INT(Y1) @ Y1
    =FP(Y1) @ Y1=10^Y1 @ IF X1<0
     THEN Y1=10*Y1
875 X1=10^X1 @ IF Y1>=2 THEN 885
880 T2=.1*X1 @ GOTO 915
885 IF Y1>=4 THEN 900
890 T2=.2*X1
895 GOTO 915
900 IF Y1>=5 THEN 910
905 T2=.4*X1 @ GOTO 915
910 T2=.5*X1
915 T5=INT(Z/T2)*T2
920 T6=0+W @ IF FP(T6/T2)#0 THEN
     T6=(INT(T6/T2)+1)*T2
925 SCALE T3-.35*P1,T4+.25*P1,T5
    -.25*P2,T6+.2*P2
930 XAXIS T5,T1,T3,T4 @ YAXIS T3
    ,T2,T5,T6
935 XAXIS T6,T1,T3,T4 @ YAXIS T4
    ,T2,T5,T6
940 G1=T3/(5*T1) @ IF FP(G1)#0 T
    HEN G1=(INT(G1)+1)*5*T1 ELSE
     G1=T3
945 I=-5
950 I=I+5 @ X=G1+I*T1
955 IF X>T4 THEN 980
960 MOVE X,T5 @ IDRAW 0,.045*P2
965 MOVE X,T6 @ IDRAW 0,-(.045*P
    2)
970 Y1=LEN(VAL$(X))
975 MOVE X-.03*(Y1-1)*P1,T5-.1*P
    2 @ LABEL VAL$(X) @ GOTO 950
980 G1=T5/(5*T2) @ IF FP(G1)#0 T
    HEN G1=(INT(G1)+1)*5*T2 ELSE
     G1=T5
985 I=-5
990 I=I+5 @ Y1=G1+I*T2 @ IF Y1>T
    6 THEN 1015
995 MOVE T3,Y1 @ IDRAW .045*P1,0
```

POLYCV HP-85

```
1000 MOVE T4,Y1 @ IDRAW -(.045*P
     1),0
1005 X=LEN(VAL$(Y1))
1010 MOVE T3-.05*(1+X)*P1,Y1-.5*
     T2 @ LABEL VAL$(Y1) @ GOTO
     990
1015 PENUP @ IF N=1 THEN 1070
1020 FOR I=1 TO N @ PLOT X(I),Y(
     I)
1025 LABEL VAL$(I)
1030 NEXT I @ PLOT X(1),Y(1)
1035 MOVE U+1.5*T1,Z-.02*P2 @ LA
     BEL "X" @ MOVE T3-.02*P1,T6
     +.085*P2 @ LABEL "Y"
1040 MOVE T3+.09*P1,T6+.1*P2
1045 IF N#1 THEN LABEL "SPECIFIE
     D POLYGON" ELSE LABEL "SPEC
     IFIED ANG.REG."
1050 C8=T3 @ C9=T5-.22*P2 @ D3=P
     1 @ GOTO 320
1055 MOVE C8-.2*D3,C9 @ LABEL "P
     ="&VAL$(I9) @ MOVE C8+.45*D
     3,C9 @ LABEL "A="&VAL$(J)
1060 MOVE C8+D3,C9 @ LABEL "W="&
     VAL$(W1) @ IF P9>1 THEN COP
     Y
1065 RETURN
1070 PLOT X(3),Y(3) @ LABEL VAL$
     (3) @ PLOT X(1),Y(1) @ LABE
     L VAL$(1)
1075 PLOT X(2),Y(2) @ LABEL VAL$
     (2) @ GOTO 1035
1080 END
```

## MLEQRE

*Input:* (Initial Data Statement), N, M, L0, A0, B0, P8

N : Number of listed successes

M : Number of listed failures

L0: If L0 = 1, initial data statement is followed by data statements containing list of successes, $a_i$, followed by data statements containing list of failures, $b_j$, i.e., $a_1$, $a_2$, ..., $a_N$, $b_1$, $b_2$, ..., $b_M$.

If L0 $\neq$ 1, initial data statement is followed by data statements containing each listed (different) success $a_i$, with each $a_i$ preceded by $c_i$, the number of times that $a_i$ occurs. Then follow data statements containing each listed (different) $b_j$ preceded by $d_j$, the number of times $b_j$ occurs, i.e., $c_1$, $a_1$, $c_2$, $a_2$, ..., $c_N$, $a_N$, $d_1$, $b_1$, $d_2$, $b_2$, ..., $d_M$, $b_M$.

The listed successes (failures) are stored by MLEQRE in Array A(I) (B(J)), I(J) = 1 to N(M). If L0 $\neq$ 1, then the $c_i(d_j)$ are stored in Array C(I) (D(J)).

If L0 = 1 then ones are stored in arrays C(I), D(J) by MLEQRE.

A0, B0: Contain initial estimates for $\alpha = \mu/\sigma$ and $\beta = 1/\sigma$ if supplied by user. If B0 $<$ 0, then initial estimates for $\alpha$ and $\beta$ are supplied by MLEQRE.

P8: If P8 = 0, no confidence plots are made. If P8 = 1, then confidence plots at the 50 and 95% level appear on the CRT. If P8 $\geq$ 2, then plots appear on CRT and also on the printer.

It is necessary and sufficient for a solution that two conditions be satisfied, namely

$$\max_j (b_j) > \min_i (a_i)$$

$$(\text{average of the } a_i) > (\text{average of the } b_j)$$

If either condition is violated an exit is made and a message is printed indicating which condition is violated.

The program is set to limit the number of iterations to 30. This number of iterations has never been necessary, however, if it should occur the constraint can be overridden as described in the comment statements at the head of MLEQRE.

*Output:* N, M. L0, A0, B0, P8

*If* (L0 = 1, (See Example 2).

| Successes | Failures |
|---|---|
| $A(1) = a_1$ | $B(1) = b_1$ |
| $A(2) = a_2$ | $B(2) = b_2$ |
| $\vdots$ | $\vdots$ |
| $A(N) = a_n$ | $B(M) = b_M$ |

**MLEQRE**

*If* (L0 ≠ 1), (See Example 1).

| NO. OF A SUCCESS AND SUCCESSES | | NO. OF A FAILURE AND FAILURES | |
|---|---|---|---|
| $c_1$ | $A(1) = a_1$ | $d_1$ | $B(1) = b_1$ |
| $c_2$ | $A(2) = a_2$ | $d_2$ | $B(2) = b_2$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $c_N$ | $A(N) = a_N$ | $d_M$ | $B(M) = b_M$ |

Maximum likelihood estimates MU and SIGMA (SIG).

Covariance elements

$a^{uu}$, $a^{us}$, $a^{ss}$

Final values $A0 = \mu/\sigma$, $B0 = 1/\sigma$.

Last Newton-Raphson corrections – (contained in D1 and D2).

Initial values for A0, B0 – (contained in A1 and B1).

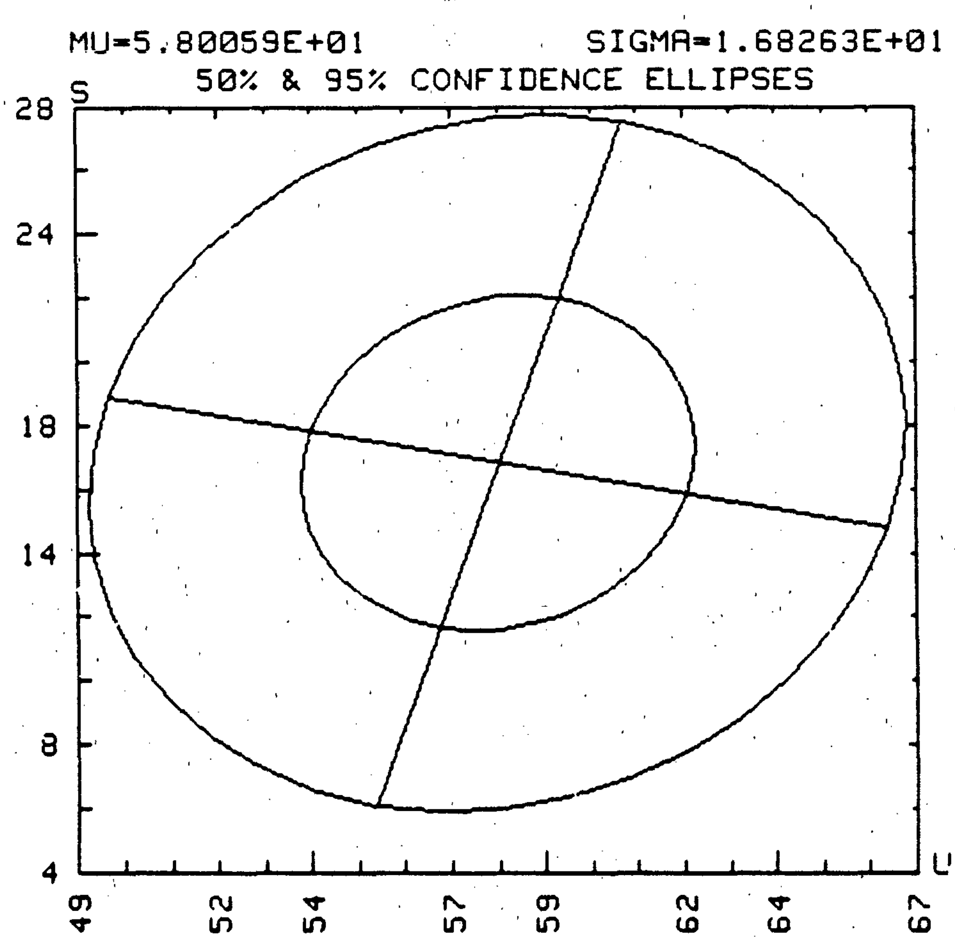
No. of N-R iterations.

This completes output if P8 = 0. If P8 = 1 or ≥2 plot follows on CRT, and in the latter case it also appears on the printer.

*Accuracy:* Approximately 6-decimal digits in $\mu$ and $\sigma$ assuming the elements of $\{a_i\}$, $\{b_j\}$ are exact.

MLEQRE--HP 9845

```
N= 4 ,M= 4 ,L0= 0 ,A0= 0 ,B0= .04 ,P8= 2

NO. OF A SUCCESS AND SUCCESSES        NO. OF A FAILURE AND FAILURES
 2       A( 1 )= 41.67                 6      B( 1 )= 27.1
 11      A( 2 )= 58.33                 9      B( 2 )= 41.67
 9       A( 3 )= 81.67                 10     B( 3 )= 58.33
 6       A( 4 )= 114.33                1      B( 4 )= 81.67

MU= 5.80059E+01   SIG= 1.68263E+01

COVARIANCE MATRIX ELEMENTS
 1.28405E+01 , 1.85908E+00 , 1.99470E+01

FINAL A0= 3.44734E+00    FINAL B0= 5.94308E-02
LAST DELTA A0,B0 = 3.79341E-06 , 6.78833E-10
INITIAL A0= 0.00000E+00   INITIAL B0= 4.00000E-02
NO. OF ITERATIONS= 7
```



**MLEQRE – HP 9845**

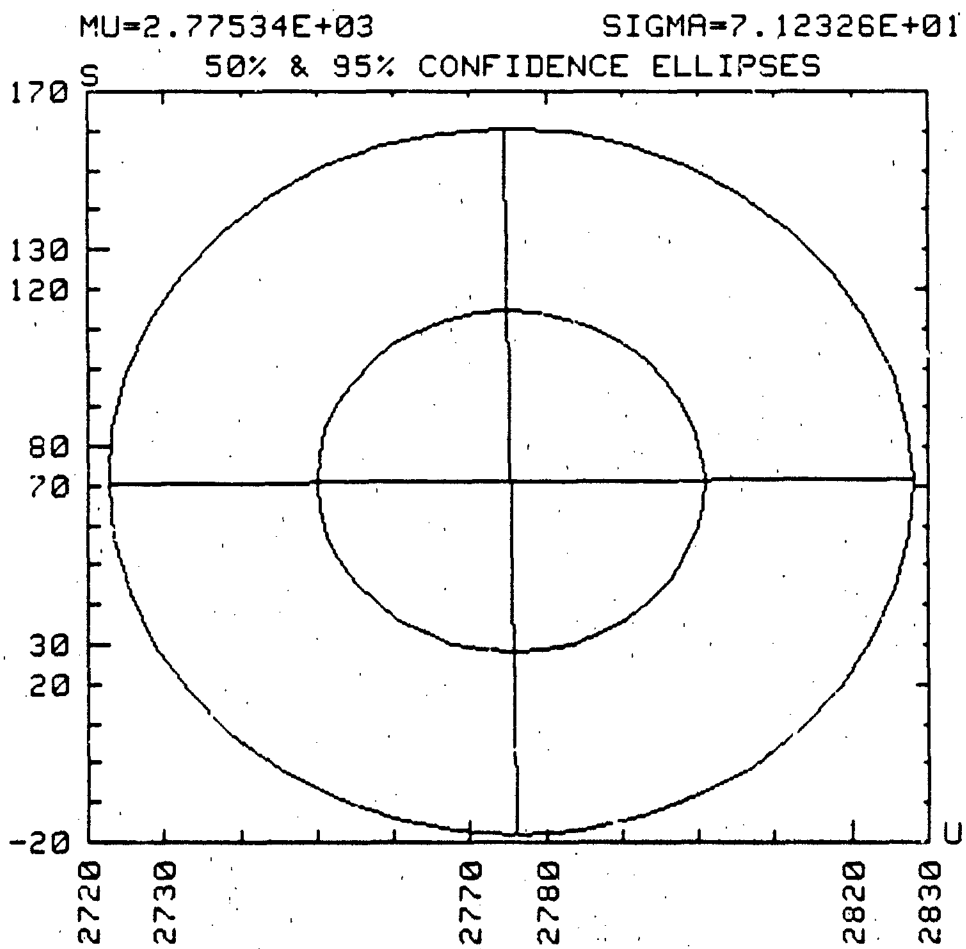
N= 10 ,M= 10 ,L0= 1 ,A0= 0 ,B0= 0 ,P8= 2

| SUCCESSES | FAILURES |
|---|---|
| A( 1 )= 2854 | B( 1 )= 2652 |
| A( 2 )= 2836 | B( 2 )= 2741 |
| A( 3 )= 2767 | B( 3 )= 2846 |
| A( 4 )= 2814 | B( 4 )= 2713 |
| A( 5 )= 2801 | B( 5 )= 2806 |
| A( 6 )= 2792 | B( 6 )= 2770 |
| A( 7 )= 2820 | B( 7 )= 2776 |
| A( 8 )= 2741 | B( 8 )= 2763 |
| A( 9 )= 2767 | B( 9 )= 2706 |
| A( 10 )= 2761 | B( 10 )= 2735 |

MU= 2.77534E+03 SIG= 7.12326E+01

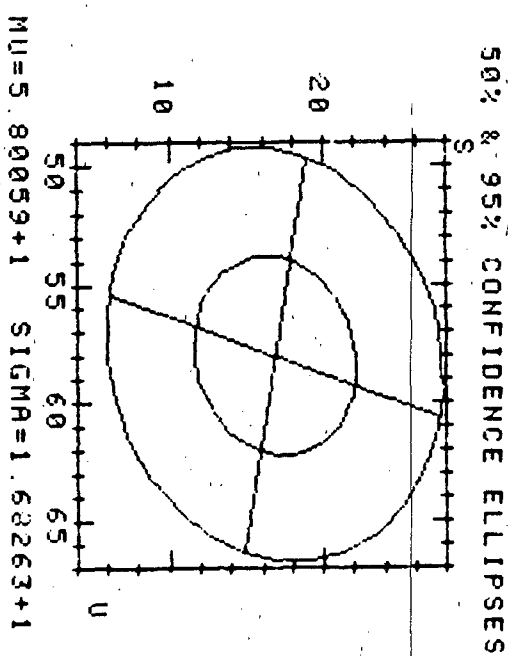
COVARIANCE MATRIX ELEMENTS
4.63975E+02 ,-7.89329E+00 , 1.33797E+03

FINAL A0= 3.89617E+01 FINAL B0= 1.40385E-02
LAST DELTA A0,B0 = 1.97272E-05 , 7.07111E-09
INITIAL A0= 5.62895E+01 INITIAL B0= 2.02988E-02
NO. OF ITERATIONS= 4



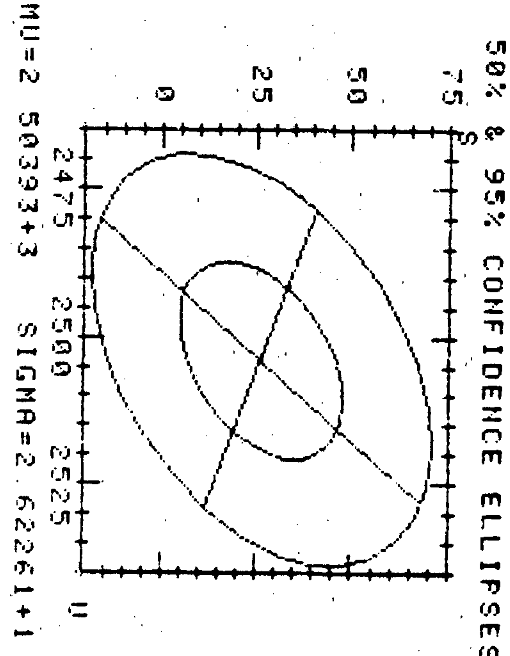
**MLEQRE – HP 85**

```
N= 4  ,M= 4  ,L0= 0  ,A0= 0  ,B0=
 .04  ,P8= 2

NO. OF A SUCCESS AND SUCCESSES
 2        A( 1 )= 41.67
 11       A( 2 )= 58.33
 9        A( 3 )= 81.67
 6        A( 4 )= 114.33

NO. OF A FAILURE AND FAILURES
 6        B( 1 )= 27.1
 9        B( 2 )= 41.67
 10       B( 3 )= 58.33
 1        B( 4 )= 81.67

MU=5.80059+1     SIG=1.68263+1

COVARIANCE MATRIX ELEMENTS
 12.8404937513   1.85908194693
 19.9470015095

INITIAL A0= 0          INITIAL B0=
 .04
FINAL A0,B0= 3.44733960086
 5.94308247597E-2
LAST DELTA A0,B0=
 3.79447746219E-8
 6.78760493686E-10
NO. OF ITERATIONS= 7
```



**MLEQRE – HP 85**

```
N= 4 ,M= 6 ,L0= 1 ,A0= 0 ,B0= 0
,PS= 2

SUCCESSES
A( 1 )= 2481
A( 2 )= 2506
A( 3 )= 2533
A( 4 )= 2620

FAILURES
B( 1 )= 2443
B( 2 )= 2486
B( 3 )= 2463
B( 4 )= 2480
B( 5 )= 2505
B( 6 )= 2500

MU=2.50393+3     SIG=2.62261+1

COVARIANCE MATRIX ELEMENTS
 202.594767688  107.937942839
 330.540527278

INITIAL A0= 54.5879207362
INITIAL B0= 2.17720294092E-2
FINAL A0,B0= 95.4744043317
 3.81298920267E-2
LAST DELTA A0,B0=
 1.6844341005E-6
 6.74520352426E-10
N0. OF ITERATIONS= 5
```



**MLEQRE – HP 9845**

```
100   ! THIS SUBROUTINE, MLEQRE, GIVES THE MAXIMUM LIKELIHOOD EST-
        IMATES,MU AND SIGMA, BASED ON QUANTAL RESPONSE EXPERIMENTS,
105   ! FOR THE MEAN AND STANDARD DEVIATION,RESPECTIVELY, OF A NORMAL
        DISTRIBUTION. OUTPUT ALSO INCLUDES THE COVARIANCE ELEMENTS
110   ! AND ,AT THE USER'S OPTION, A PLOT OF THE CONFIDENCE ELLIPSES
        AT THE 50 AND 95% LEVELS. THE QUANTAL RESPONSES ARE CLASSIFIED
115   ! AS "SUCCESSES" OR "FAILURES". THE INPUT QUANTITIES TO MLEQRE
        RESULTING IN SUCCESSES(FAILURES) ARE ALSO SIMPLY REFERRED TO
120   ! AS SUCCESSES (FAILURES).
125   ! THE INPUT IS STORED IN A DATA STATEMENT BEGINNING WITH:
        N,M,L0,A0,B0,P8 WHERE N(M) DENOTES THE NUMBER OF "LISTED"
130   ! SUCCESSES(FAILURES). IF L0=1, THEN THE DATA STATEMENT IS
        CONTINUED WITH THE LIST OF SUCCESSES FOLLOWED BY THE LIST
135   ! OF FAILURES. IF L0#1, THEN EACH DIFFERENT SUCCESS IS PRECEDED
        BY THE NO. OF TIMES IT OCCURS. THE SAME IS THEN DONE WITH
140   ! FAILURES. THE LISTED SUCCESSES(FAILURES) ARE STORED BY MLEQRE
        IN A(I)(B(J)), I(J)=1 TO N(M). IF L0#1 THEN THE NO. OF A(I)
145   ! (B(J)) AT A FIXED I(J) FOR EACH I(J) IS STORED BY MLEQRE IN
        C(I)(D(J)). IF L0=1, THEN C(I)(D(J)) CONTAINS N(M) ONES.
150   ! A0 AND  B0  CONTAIN INITIAL ESTIMATES FOR MU/SIGMA
        AND 1/SIGMA. IF B0<=0, THEN A0 AND B0 ARE SUPPLIED BY MLEQRE.
155   ! MAXIMUM NO. OF DIFFERENT A(I) IS 100. SIMILARLY FOR THE B(J).
        TO INCREASE, CHANGE DIMENSION STATEMENT AT LINE 345.
160   ! P8 IS AN OUTPUT SPECIFIER. IF P8=0, THEN NO PLOT IS MADE. IF
        P8=1, THEN PLOT APPEARS ON CRT. IF P8>=2, THEN PLOT APPEARS ON
165   ! CRT AND PRINTER.
170   ! IT IS NECESSARY AND SUFFICIENT FOR A SOLUTION TO EXIST THAT
        MAX. LISTED FAILURE>MIN. LISTED SUCCESS AND THAT THE AVERAGE
175   ! OF THE SUCCESSES>THE AVERAGE OF THE FAILURES. IF EITHER COND-
        ITION IS VIOLATED AN ERROR MESSAGE IS PRINTED FROM LINE 710
180   ! OR 735. A MAX. OF 30 ITERATIONS IS ALLOWED. MESSAGE PRINTED
        FROM LINE 800. IF MORE DESIRED CHANGE LINE 790 AND CONTINUE
185   ! AT LINE 770.
190   ! SOURCES: NWL REPORT 2846, NOV. 1972. SIAM JN.
        OF APPL. MATH., MAR. 1972, PP. 447-454.
195   OPTION BASE 1
200   DIM P9(4),Q9(3),R9(6),S9(5),V9(4),W9(3),D$[30],E$[29]
205   P9(1)=-3.56098437018E-2
210   P9(2)=6.99638348862
215   P9(3)=21.9792616183
220   P9(4)=242.667955231
225   Q9(1)=15.0827976304
230   Q9(2)=91.1649054045
235   Q9(3)=215.05887587
240   R9(1)=-6.08581519597E-6
245   R9(2)=.564371606854
250   R9(3)=4.26772010709
255   R9(4)=14.5718985969
260   R9(5)=26.0947469561
265   R9(6)=22.8989928517
270   S9(1)=7.56884822936
275   S9(2)=26.2887957588
280   S9(3)=50.2732028638
```

```
285   S9(4)=51.9335706876
290   S9(5)=22.8989857499
295   V9(1)=-3.24319519278E-2
300   V9(2)=-.243911029489
305   V9(3)=-.119903955268
310   V9(4)=-.012130827639
315   W9(1)=1.43771227937
320   W9(2)=.489552441961
325   W9(3)=4.30026643453E-2
330   C1=.707106781187  !1/SQR(2)
335   C2=.797884560803  !SQR(2/PI)
340   C3=.564189583548  !SQR(1/PI)
345   DIM A(100),B(100),C(100),D(100)
350   D$="NO. OF A SUCCESS AND SUCCESSES"
355   E$="NO. OF A FAILURE AND FAILURES"
360   STANDARD
365   READ N,M,L0,A0,B0,P8
370   PRINT "N=";N;",";"M=";M;",";"L0=";L0;",";"A0=";A0;",";"B0=";B0;",";"P8=";P
8
375   PRINT
380   IF L0=1 THEN 485
385   PRINT D$;TAB(38);E$
390   FOR I=1 TO N
395     READ C(I),A(I)
400   NEXT I
405   FOR J=1 TO M
410     READ D(J),B(J)
415   NEXT J
420   FOR I=1 TO MIN(N,M)
425     PRINT C(I);TAB(9);"A(";I;")=";A(I);TAB(38);D(I);TAB(45);"B(";I;")=";B(I)
430   NEXT I
435   IF M>N THEN 465
440   IF M=N THEN 590
445   FOR I=M+1 TO N
450     PRINT C(I);TAB(9);"A(";I;")=";A(I)
455   NEXT I
460   GOTO 590
465   FOR I=N+1 TO M
470     PRINT TAB(38);D(I);TAB(45);"B(";I;")=";B(I)
475   NEXT I
480   GOTO 590
485   PRINT D$[22,30];TAB(30);E$[22,29]
490   FOR I=1 TO N
495     READ A(I)
500     C(I)=1
505   NEXT I
510   FOR J=1 TO M
515     READ B(J)
520     D(J)=1
525   NEXT J
530   FOR I=1 TO MIN(M,N)
535     PRINT "A(";I;")=";A(I);TAB(30);"B(";I;")=";B(I)
540   NEXT I
```

```
545    IF M>N THEN 575
550    IF M=N THEN 590
555    FOR I=M+1 TO N
560      PRINT "A(";I;")=";A(I)
565    NEXT I
570    GOTO 590
575    FOR I=N+1 TO M
580      PRINT TAB(30);"B(";I;")=";B(I)
585    NEXT I
590    ! PAUSE
595    Z=1E99
600    D=-Z
605    E1=.008
610    E2=.01
615    L1=L2=L3=0
620    FOR I=1 TO N
625      L1=L1+C(I)
630      IF A(I)<Z THEN Z=A(I)
635      P1=C(I)*A(I)
640      L2=L2+P1
645      P2=P1*A(I)
650      L3=L3+P2
655    NEXT I
660    L4=L5=L6=0
665    FOR J=1 TO M
670      L4=L4+D(J)
675      IF B(J)>D THEN D=B(J)
680      P3=D(J)*B(J)
685      L5=L5+P3
690      P4=P3*B(J)
695      L6=L6+P4
700    NEXT J
705    IF Z<D THEN 720
710    PRINT "A-MIN>=B-MAX"
715    RETURN
720    L7=L2/L1
725    L8=L5/L4
730    IF L7>L8 THEN 745
735    PRINT "AVE. OF A's <= AVE. OF B's"
740    RETURN
745    IF B0>0 THEN 770
750    L1=L1+L4
755    V=(L2+L5)/L1
760    B0=SQR(1/((L3+L6)/L1-V*V))
765    A0=.5*(L7+L8)*B0
770    A1=A0
775    B1=B0
780    Z=L6=L7=L8=L9=0
785    L1=L2=L3=L4=L5=0
790    IF L9<30 THEN 810
795    U=1E99
800    PRINT "30 ITERATIONS"
805    RETURN
```

```
810   L9=L9+1
815   FOR I=1 TO N
820     S=B0*A(I)-A0
825     T=-S
830     GOSUB 1215
835     P=C(I)*Y
840     L1=L1-P
845     P1=A(I)*P
850     L2=L2+P1
855     P2=C(I)*Y1
860     L3=L3-P2
865     P3=A(I)*P2
870     L4=L4+P3
875     P4=A(I)*P3
880     L5=L5-P4
885     IF Z=0 THEN 920
890     P5=C(I)*Y2
895     L6=L6+P5
900     P6=S*P5
905     L7=L7+P6
910     P7=S*P6
915     L8=L8+P7
920   NEXT I
925   FOR J=1 TO M
930     T=B0*B(J)-A0
935     GOSUB 1215
940     P=D(J)*Y
945     L1=L1+P
950     P1=B(J)*P
955     L2=L2-P1
960     P2=D(J)*Y1
965     L3=L3-P2
970     P3=B(J)*P2
975     L4=L4+P3
980     P4=B(J)*P3
985     L5=L5-P4
990     IF Z=0 THEN 1025
995     P5=D(J)*Y2
1000    L6=L6+P5
1005    P6=T*P5
1010    L7=L7+P6
1015    P7=T*P6
1020    L8=L8+P7
1025  NEXT J
1030  D=L3*L5-L4*L4
1035  D1=(L2*L4-L1*L5)/D
1040  D2=(L1*L4-L2*L3)/D
1045  A0=A0+D1
1050  B0=B0+D2
1055  IF (ABS(D1)>E1*(ABS(A0)+.00000001)) OR (ABS(D2)>E2*B0) THEN 785
1060  Z=Z+1
1065  IF Z<2 THEN 785
1070  S=1/B0  !S=SIGMA
```

```
1075  U=A0÷S  !U=MU
1080  ! L6=Auu, L7=Aus, L8=Ass
1085  Z=B0*B0
1090  L6=L6*Z
1095  L7=L7*Z
1100  L8=L8*Z
1105  L0=L6*L8-L7*L7
1110  L1=1/L0
1115  L2=L8*L1
1120  L3=-L7*L1
1125  L4=L6*L1
1130  ! L2,L3,L4 ARE THE COVARIANCE MATRIX ELEMENTS.
1135  PRINT
1140  FLOAT 5
1145  PRINT "MU=";U;TAB(19);"SIG=";S
1150  PRINT
1155  PRINT "COVARIANCE MATRIX ELEMENTS "
1160  PRINT L2;",";L3;",";L4
1165  PRINT
1170  PRINT "FINAL A0=";A0;TAB(26);"FINAL B0=";B0
1175  PRINT "LAST DELTA A0,B0 =";D1;",";D2
1180  PRINT "INITIAL A0=";A1;TAB(27);"INITIAL B0=";B1
1185  STANDARD
1190  PRINT "NO. OF ITERATIONS=";L9
1195  PRINT
1200  IF P8>0 THEN 1440 ! FOR CONFIDENCE ELLIPSE PLOTS.
1205  RETURN
1210  ! CODY. INPUT:T. OUTPUT:Y=SQR(2/PI)*EXP(-K1*K1)/ERFC(K1), Y1=Y*(Y-T),
        Y2= SQR(2/PI)*EXP(-K1*K1)*Y/(2-ERFC(K1)). 12-DECIMAL ACCURACY.
1215  K1=T*C1
1220  Y=Y1=Y2=0
1225  IF K1<=-5.5 THEN RETURN
1230  IF ABS(K1)>.5 THEN 1275
1235  K4=K1*K1
1240  K3=(((P9(1)*K4+P9(2))*K4+P9(3))*K4+P9(4))/(((K4+Q9(1))*K4+Q9(2))*K4+Q9(3))
1245  Y=1-K1*K3
1250  W=C2*EXP(-K4)
1255  Y=W/Y
1260  Y1=Y*(Y-T)
1265  IF Z<>0 THEN Y2=W*Y/(1+K1*K3)
1270  RETURN
1275  K4=ABS(K1)
1280  IF K4>4 THEN 1350
1285  K3=(((((R9(1)*K4+R9(2))*K4+R9(3))*K4+R9(4))*K4+R9(5))*K4+R9(6))/(((((K4+S9
(1))*K4+S9(2))*K4+S9(3))*K4+S9(4))*K4+S9(5))
1290  IF K1<0 THEN 1320
1295  Y=C2/K3
1300  IF Z=0 THEN 1340
1305  W=EXP(-K1*K1)
1310  Y2=C2*W*Y/(2-W*K3)
1315  GOTO 1340
1320  W=EXP(-K1*K1)
1325  Y=C2*W/(2-W*K3)
1330  IF Z=0 THEN 1340
```

```
1335  Y2=C2*Y/K3
1340  Y1=Y*(Y-T)
1345  RETURN
1350  K6=1/(K1*K1)
1355  K3=(((V9(1)*K6+V9(2))*K6+V9(3))*K6+V9(4))/(((K6+W9(1))*K6+W9(2))*K6+W9(3))
1360  W1=C3+K3*K6
1365  IF K1<0 THEN 1400
1370  Y=C2*K1/W1
1375  Y1=-(2*C3*K3/(W1*W1))
1380  IF (Z=0) OR (K1>5.5) THEN RETURN
1385  W=EXP(-K1*K1)
1390  Y2=C2*W*Y/(2-W*W1/K1)
1395  RETURN
1400  W=EXP(-K1*K1)
1405  Y=C2*W/(2-W*W1/K4)
1410  Y1=Y*(Y-T)
1415  IF Z=0 THEN RETURN
1420  Y2=C2*K4*Y/W1
1425  RETURN
1430  Y=Y1=Y2=0
1435  RETURN
1440  EXIT GRAPHICS  ! PLOTTING OF CONFIDENCE ELLIPSES FOLLOWS.
1445  PLOTTER IS "GRAPHICS"
1450  GRAPHICS
1455  L9=5.99  !L9= 95% CHI-SQUARED VALUE.
1460  S1=2*SQR(L8*L9/L0)  ! RANGE OF X-VALUES.
1465  S2=2*SQR(L6*L9/L0)  ! RANGE OF Y-VALUES.
1470  Y=LGT(S1)
1475  T=INT(Y)
1480  Y=FRACT(Y)
1485  Y=10^Y
1490  IF T<0 THEN Y=10*Y
1495  T=10^T
1500  IF Y>=2 THEN 1515
1505  K1=.1*T
1510  GOTO 1550
1515  IF Y>4 THEN 1530
1520  K1=.2*T
1525  GOTO 1550
1530  IF Y>=5 THEN 1545
1535  K1=.4*T
1540  GOTO 1550
1545  K1=.5*T
1550  K3=INT((U-.5*S1)/K1)*K1
1555  K4=U+.5*S1
1560  IF FRACT(K4/K1)<>0 THEN K4=(INT(K4/K1)+1)*K1
1565  Y=LGT(S2)
1570  T=INT(Y)
1575  Y=FRACT(Y)
1580  Y=10^Y
1585  IF T<0 THEN Y=10*Y
1590  T=10^T
1595  IF Y>=2 THEN 1610
```

```
1600  K2=.1*T
1605  GOTO 1645
1610  IF Y>=4 THEN 1625
1615  K2=.2*T
1620  GOTO 1645
1625  IF Y>=5 THEN 1640
1630  K2=.4*T
1635  GOTO 1645
1640  K2=.5*T
1645  K5=INT((S-.5*S2)/K2)*K2
1650  K6=S+.5*S2
1655  IF FRACT(K6/K2)<>0 THEN K6=(INT(K6/K2)+1)*K2
1660  SCALE K3-.35*S1,K4+.25*S1,K5-.25*S2,K6+.2*S2
1665  CLIP K3,K4,K5,K6
1670  LAXES K1,K2,K3,K5,5,5,4
1675  LAXES K1,K2,K4,K6,5,5,2
1680  !  ELLIPSE PLOT
1685  L1=C1
1690  L2=SGN(L7)*C1
1695  P1=L6-L8
1700  IF ABS(P1)<(ABS(L6)+ABS(L8))*1E-10 THEN 1725
1705  P3=ATN(2*L7/P1)
1710  P3=.5*P3
1715  L1=COS(P3)
1720  L2=SIN(P3)
1725  I=0
1730  K=40
1735  P5=PI+PI
1740  P7=P5/K   !K= NO. OF POINTS FOR ELLIPSE.
1745  X=L1*L1
1750  Y=L2*L2
1755  T=SQR(L9/(L6*X+L8*Y+2*L7*L1*L2))
1760  Y1=SQR(L9/(L6*Y+L8*X-2*L7*L1*L2))
1765  X=COS(P7)
1770  Y=SIN(P7)
1775  P3=T*L2
1780  P4=Y1*L1
1785  P1=T*L1
1790  P2=Y1*L2
1795  Z=-P7
1800  W=1
1805  Y2=0
1810  MOVE U+P1,S+P3
1815  Z=Z+P7
1820  IF Z>P5 THEN 1850
1825  P6=W*X-Y2*Y
1830  Y2=W*Y+Y2*X
1835  W=P6
1840  DRAW U+P1*W-P2*Y2,S+P3*W+P4*Y2
1845  GOTO 1815
1850  IF I<>0 THEN 1900
1855  I=1
1860  MOVE U+P1,S+P3
```

```
365   DRAW U-P1,S-P3
370   MOVE U+P2,S-P4
375   DRAW U-P2,S+P4
380   I8=SQR(1.39/L9)  !1.39 IS 50% CHI-SQUARED VALUE.
385   T=T*I8
390   Y1=Y1*I8
395   GOTO 1775
900   MOVE K3+.15*S1,K6+.025*S2
905   LABEL "50% & 95% CONFIDENCE ELLIPSES"
910   FLOAT 5
915   MOVE K3-.005*S1,K6+.08*S2
920   LABEL "MU="&VAL$(U)
925   MOVE K4-.4*S1,K6+.08*S2
930   LABEL "SIGMA="&VAL$(S)
935   MOVE K4+.02*S1,K5
940   LABEL "U"
945   MOVE K3-.005*S1,K6+.01*S2
950   LABEL "S"
955   IF P8>=2 THEN DUMP GRAPHICS
960   RETURN
```

MLEQRE – HP 85

```
100 ! THIS SUBROUTINE, MLEQRE, G
    IVES THE MAXIMUM LIKELIHOOD
    ESTIMATES, MU AND SIGMA, BAS
    ED ON
105 ! QUANTAL RESPONSE EXPERIMEN
    TS, FOR THE MEAN AND STANDAR
    D DEVIATION,RESPECTIVELY, OF
     A NORMAL
110 ! DISTRIBUTION. OUTPUT ALSO
    INCLUDES THE COVARIANCE ELEM
    ENTS AND ,AT THE USER'S OPTI
    ON, A PLOT
115 ! OF THE CONFIDENCE ELLIPSES
     AT THE 50 AND 95% LEVELS. T
    HE QUANTAL RESPONSES ARE CLA
    SSIFIED
120 ! AS "SUCCESSES" OR "FAILURE
    S". THE INPUT QUANTITIES TO
    MLEQRE RESULTING FROM SUCCES
    SES(
125 ! FAILURES) ARE ALSO SIMPLY
    REFERRED TO AS SUCCESSES(FAI
    LURES). THE INPUT IS STORED
    IN A
130 ! DATA STATEMENT(S) BEGINNIN
    G WITH N,M,L0,A0,B0,P8,WHERE
     N(M) DENOTES
135 ! THE NUMBER OF "LISTED" SUC
    CESSES(FAILURES). IF L0=1, T
    HEN THE DATA STATEMENT IS CO
    NTINUED
140 ! WITH THE LIST OF SUCCESSES
     FOLLOWED BY THE LIST OF FAI
    LURES. IF L0#1, THEN EACH DI
    FFERENT
145 ! SUCCESS IS PRECEDED BY THE
     NO. OF TIMES IT OCCURS. THE
     SAME IS THEN DONE WITH THE
    FAILURES.
150 ! THE LISTED SUCCESSES(FAILU
    RES) ARE STORED BY MLEQRE IN
     A(I)(B(J)), I(J)=1 TO N(M).
     IF L0#1
155 ! THEN THE NO. OF A(I)(B(J))
     AT A FIXED I(J) FOR EACH I(
    J) IS STORED BY MLEQRE IN C(
    I)(D(J))
160 ! IF L0=1, THEN C(I)(D(J)) C
    ONTAINS N(M) ONES.
165 ! A0 AND B0 CONTAIN THE INIT
    IAL ESTIMATES FOR MU/SIGMA A
    ND 1/SIGMA, RESPECTIVELY
170 ! IF B0<=0, THEN MLEQRE SUPP
    LIES THE INITIAL ESTIMATES F
    OR A0 AND B0.
175 ! THE A(I),B(J),C(I),D(J) AR
    E DIMENSIONED 100 AT LINE 27
    5.
180 ! P8 IS AN OUTPUT SPECIFIER.
    IF P8=0,THEN NO PLOT IS MADE
    . IF P8=1,THEN PLOT APPEARS
    ON CRT.
185 ! IF P8>=2, THEN PLOT APPEAR
    S ON CRT AND ON THE PRINTER.
190 ! IT IS NECESSARY AND SUFFIC
    IENT FOR A SOLUTION TO EXIST
     THAT THE MAX. FAILURE>MIN.
    SUCCESS,
195 ! AND THAT THE AVERAGE OF TH
    E SUCCESSES>AVERAGE OF THE F
    AILURES. IF EITHER CONDITION
     IS VIO-
200 ! LATED AN ERROR MESSAGE IS
    PRINTED FROM LINES 440 OR 45
    0. A MAX.OF 30 ITERATIONS IS
     ALLOWED
205 ! IF MORE DESIRED,CHANGE 30
    IN LINE 480 AND CONTINUE AT
    470.
210 ! SOURCES:NWL REPORT 2846,NO
    V.1972@SIAM JN. OF APPL. MAT
    H.,  MAR.1972,PP.447,454.
215 OPTION BASE 1
220 DIM P9(4),Q9(3),R9(6),S9(5),
    V9(4),W9(3),D$[30],E$[29]
225 P9(1)=-3.56098437013E-2 @ P9
    (2)=6.99638348862 @ P9(3)=21
    .9792616183 @ P9(4)=242.5679
    55231
230 Q9(1)=15.0827976304 @ Q9(2)=
    91.1649054045 @ Q9(3)=215.05
    887587
235 R9(1)=-6.08581519597E-6 @ R9
    (2)=.564371606864 @ R9(3)=4.
    26772010709
240 R9(4)=14.5718985969 @ R9(5)=
    26.0947460561 @ R9(6)=22.898
    9928517
245 S9(1)=7.56884822936 @ S9(2)=
    26.2887957588
250 S9(3)=50.2732028638 @ S9(4)=
    51.9335706876 @ S9(5)=22.898
    9857499
255 V9(1)=-3.24319519278E-2 @ V9
    (2)=-.243911029489 @ V9(3)=-
    .119903955268 @ V9(4)=-.0121
    30827639
260 W9(1)=1.43771227937 @ W9(2)=
    .489552441961 @ W9(3)=4.3002
    6643453E-2
```

MLEQRE – HP 85

```
C1=.707106781187 @ C2=.79788
4560803 @ C3=.564189583548 @
! C2=SQR(2/PI)@C3=SQR(1/PI)
! INPUT N,M,L0,A0,B0
DIM A(100),C(100),B(100),D(1
00)
READ N,M,L0,A0,B0,P8
PRINT "N=";N;",";"M=";M;",";
"L0=";;L0;",";"A0=";A0;",";"
B0=";B0;",";"P8=";P8 @ PRINT
D$="NO. OF A SUCCESS AND SUC
CESSES" @ E$="NO. OF A FAILU
RE AND FAILURES"
IF L0=1 THEN 350
PRINT D$
FOR I=1 TO N
READ C(I),A(I)
PRINT C(I);TAB(9);"A(";I;")=
";A(I)
NEXT I
PRINT @ PRINT E$
FOR J=1 TO M
READ D(J),B(J)
PRINT D(J);TAB(9);"B(";J;")=
";B(J) @ NEXT J
GOTO 380
PRINT D$[22,30]
FOR I=1 TO N
READ A(I)@ PRINT "A(";I;")="
;A(I) @ C(I)=1 @ NEXT I
PRINT @ PRINT E$[22,29]
FOR J=1 TO M
READ B(J)@ PRINT "B(";J;")="
;B(J) @ D(J)=1 @ NEXT J
Z=1.E99 @ D=-Z @ E1=.008 @ E
2=.01 @ LET L1,L2,L3=0
FOR I=1 TO N
L1=L1+C(I) @ IF A(I)<Z THEN
Z=A(I)
P1=C(I)*A(I) @ L2=L2+P1
P2=P1*A(I) @ L3=L3+P2
NEXT I
L4=0 @ L5=0 @ L6=0
FOR J=1 TO M
L4=L4+D(J) @ IF B(J)>D THEN
D=B(J)
P3=D(J)*B(J) @ L5=L5+P3
P4=P3*B(J) @ L6=L6+P4
NEXT J
IF Z>=D THEN PRINT "A-MIN>=B
-MAX" ELSE 450
RETURN
L7=L2/L1 @ L8=L5/L4 @ IF L7<
=L8 THEN PRINT "AVE OF THE A
'S<=AVE OF THE B'S" ELSE 460
455 RETURN
460 IF B0>0 THEN 470
465 L1=L1+L4 @ V=(L2+L5)/L1 @ B0
    =SQR(1/((L3+L6)/L1-V*V)) @ A
    0=.5*(L7+L8)*B0
470 A1=A0 @ B1=B0
475 LET Z,L6,L7,L8,L9=0
480 LET L1,L2,L3,L4,L5=0 @ IF L9
    <30 THEN L9=L9+1 ELSE U=1.E9
    9 @ PRINT "30 ITERATIONS" @
    RETURN
485 FOR I=1 TO N
490 S=B0*A(I)-A0 @ T=-S
495 GOSUB 695
500 P=C(I)*Y @ L1=L1-P
505 P1=A(I)*P @ L2=L2+P1
510 P2=C(I)*Y1 @ L3=L3-P2
515 P3=A(I)*P2 @ L4=L4+P3
520 P4=A(I)*P3 @ L5=L5-P4
525 IF Z=0 THEN 545
530 P5=C(I)*Y2 @ L6=L6+P5
535 P6=S*P5 @ L7=L7+P6
540 P7=S*P6 @ L8=L8+P7
545 NEXT I
550 FOR J=1 TO M
555 T=B0*B(J)-A0
560 GOSUB 695
565 P=D(J)*Y @ L1=L1+P
570 P1=B(J)*P @ L2=L2-P1
575 P2=D(J)*Y1 @ L3=L3-P2
580 P3=B(J)*P2 @ L4=L4+P3
585 P4=B(J)*P3 @ L5=L5-P4
590 IF Z=0 THEN 610
595 P5=D(J)*Y2 @ L6=L6+P5
600 P6=T*P5 @ L7=L7+P6
605 P7=T*P6 @ L8=L8+P7
610 NEXT J
615 D=L3*L5-L4*L4 @ D1=(L2*L4-L1
    *L5)/D @ D2=(L1*L4-L2*L3)/D
620 A0=A0+D1 @ B0=B0+D2
625 IF ABS(D1)>E1*(ABS(A0)+.0000
    0001) OR ABS(D2)>E2*B0 THEN
    480
630 Z=Z+1 @ IF Z#2 THEN 480
635 S=1/B0 @ U=A0*S
640 ! U=MU@S=SIGMA@L6=A'SUB MUMU
    '@L7=A'SUB MUS'@L8=A'SUBSS'.
645 Z=B0*B0 @ L6=L6*Z @ L7=L7*Z
    @ L8=L8*Z @ L0=L6*L8-L7*L7 @
     L1=1/L0 @ L2=L8*L1 @ L3=-(L
    7*L1) @ L4=L6*L1
653 ! L2,L3,L4 ARE THE ELEMENTS
    OF THE COVARIANCE MATRIX
655 PRINT @ Z1=U @ GOSUB 1160
660 B$=A$ @ Z1=S @ GOSUB 1160
```

**MLEQRE – HP 85**

```
65 PRINT "MU=";B$;TAB(17);"SIG=
   ";A$ @ PRINT
70 PRINT "COVARIANCE MATRIX ELE
   MENTS" @ PRINT L2;L3;L4 @ PR
   INT
75 PRINT "INITIAL A0=";A1,"INIT
   IAL B0=";B1,"FINAL A0,B0=";A
   0,B0,"LAST DELTA A0,B0=";D1;
   D2
680 PRINT "NO. OF ITERATIONS=";L
   9 @ PRINT @ IF P8#0 THEN 825
    ELSE RETURN ! CONFIDENCE PL
   OTS.
685 REM CODY. INPUT T. OUTPUT:
    Y(K1)=SQR(2/PI)*EXP(-K1*K1)
   /ERFC(K1), Y1=Y*(Y-T)
690 ! OUTPUT CONTINUED:Y2=SQR(2/
   PI)*EXP(-K1*K1)*Y/(2-ERFC)
695 K1=T*C1 @ Y=0 @ Y1=0 @ Y2=0
   @ IF K1<=-5.5 THEN RETURN
700 IF ABS(K1)>.5 THEN 730
705 K4=K1*K1
710 K3=(((P9(1)*K4+P9(2))*K4+P9(
   3))*K4+P9(4))/(((K4+Q9(1))*K
   4+Q9(2))*K4+Q9(3))
715 Y=1-K1*K3 @ W=C2*EXP(-K4) @
   Y=W/Y @ Y1=Y*(Y-T)
720 IF Z#0 THEN Y2=W*Y/(1+K1*K3)
725 RETURN
730 K4=ABS(K1) @ IF K4>4 THEN 77
   5
735 K3=(((((R9(1)*K4+R9(2))*K4+R9
   (3))*K4+R9(4))*K4+R9(5))*K4+
   R9(6)
740 K3=K3/(((((K4+S9(1))*K4+S9(2
   ))*K4+S9(3))*K4+S9(4))*K4+S9
   (5))
745 IF K1>=0 THEN Y=C2/K3 ELSE 7
   60
750 IF Z#0 THEN W=EXP(-(K1*K1))
   ELSE 770
755 Y2=C2*W*Y/(2-W*K3) @ GOTO 77
   0
760 W=EXP(-(K1*K1)) @ Y=C2*W/(2-
   W*K3)
765 IF Z#0 THEN Y2=C2*Y/K3
770 Y1=Y*(Y-T) @ RETURN
775 K6=1/(K1*K1)
780 K3=(((V9(1)*K6+V9(2))*K6+V9(
   3))*K6+V9(4))/(((K6+W9(1))*K
   6+W9(2))*K6+W9(3))
785 W1=C3+K6*K3 @ IF K1<0 THEN 8
   10
790 Y=C2*K1/W1 @ Y1=-(2*C3*K3/(W
   1*W1))
795 IF Z#0 THEN W=EXP(-(K1*K1))
   ELSE RETURN
800 Y2=C2*W*Y/(2-W*W1/K1) @ RETU
   RN
805 Y=0 @ Y1=0 @ Y2=0 @ RETURN
810 W=EXP(-(K1*K1)) @ Y=C2*W/(2-
   W*W1/K4) @ Y1=Y*(Y-T)
815 IF Z#0 THEN Y2=Y*C2*K4/W1
820 RETURN
825 ! PLOTTING BEGINS
830 GCLEAR @ L9=5.99 ! L9=95%CHI
   -SQUARED.
835 S1=2*SQR(L8*L9/L0) @ S2=2*SQ
   R(L6*L9/L0)
840 Y=LGT(S1) @ T=INT(Y) @ Y=FP(
   Y)
845 IF T<0 THEN Y=10^(Y+1) ELSE
   Y=10^Y
850 T=10^T
855 IF Y>=2 THEN 865
860 K1=.1*T @ GOTO 890
865 IF Y>=4 THEN 875
870 K1=.2*T @ GOTO 890
875 IF Y>=5 THEN 885
880 K1=.4*T @ GOTO 890
885 K1=.5*T
890 K3=INT((U-.5*S1)/K1)*K1
895 K4=U+.5*S1 @ IF FP(K4/K1)#0
   THEN K4=(INT(K4/K1)+1)*K1
900 Y=LGT(S2) @ T=INT(Y) @ Y=FP(
   Y)
905 IF T<0 THEN Y=10^(Y+1) ELSE
   Y=10^Y
910 T=10^T
915 IF Y>=2 THEN 925
920 K2=.1*T @ GOTO 950
925 IF Y>=4 THEN 935
930 K2=.2*T @ GOTO 950
935 IF Y>=5 THEN 945
940 K2=.4*T @ GOTO 950
945 K2=.5*T
950 K5=INT((S-.5*S2)/K2)*K2
955 K6=S+.5*S2 @ IF FP(K6/K2)#0
   THEN K6=(INT(K6/K2)+1)*K2
960 SCALE K3-.35*S1,K4+.25*S1,K5
   -.25*S2,K6+.2*S2
965 XAXIS K5,K1,K3,K4 @ YAXIS K3
   ,K2,K5,K6
970 XAXIS K6,K1,K3,K4 @ YAXIS K4
   ,K2,K5,K6
975 W1=K3/(5*K1) @ IF FP(W1)#0 T
   HEN W1=(INT(W1)+1)*5*K1 ELSE
    W1=K3
980 J=-5
```

```
J=J+5 @ X=W1+J*K1 @ IF X>K4
THEN 1010
MOVE X,K5 @ IDRAW 0,.045*S2
MOVE X,K6 @ IDRAW 0,-(.045*S
2)
Y=LEN(VAL$(X))
MOVE X-.03*(Y-1)*S1,K5-.1*S
2 @ LABEL VAL$(X) @ GOTO 98
5
W1=K5/(5*K2) @ IF FP(W1)#0
THEN W1=(INT(W1)+1)*5*K2 EL
SE W1=K5
J=-5
J=J+5 @ Y=W1+J*K2 @ IF Y>K6
THEN 1050
MOVE K3,Y @ IDRAW .045*S1,0
MOVE K4,Y @ IDRAW -(.045*S1
),0
X=LEN(VAL$(Y))
MOVE K3-.05*(1+X)*S1,Y-.5*K
2 @ LABEL VAL$(Y) @ GOTO 10
20
! ELLIPSE PLOTS
L1=C1 @ L2=SGN(L7)*C1 @ P1=
L6-L8 @ IF ABS(P1)<(ABS(L6)
+ABS(L8))*.0000000001 THEN
1060
P3=.5*ATN(2*L7/P1) @ L1=COS
(P3) @ L2=SIN(P3)
I=0 @ K=30 @ P5=PI+PI @ P7=
P5/K ! K= N0. OF POINTS FOR
PLOTS OF ELLIPSIS.
X=L1*L1 @ Y=L2*L2 @ T=SQR(L
9/(L6*X+L8*Y+2*L7*L1*L2)) @
Y1=SQR(L9/(L6*Y+L8*X-2*L7*
L1*L2))
X=COS(P7) @ Y=SIN(P7)
P3=T*L2 @ P4=Y1*L1 @ P1=T*L
1 @ P2=Y1*L2
Z=-P7 @ W=1 @ Y2=0
MOVE U+P1,S+P3
Z=Z+P7 @ IF Z>P5 THEN 1105
P6=W*X-Y2*Y @ Y2=W*Y+Y2*X @
W=P6 @ DRAW U+P1*W-P2*Y2,S
+P3*W+P4*Y2
GOTO 1090
1105 IF I#0 THEN 1120 ELSE I=1
1110 MOVE U+P1,S+P3 @ DRAW U-P1,
     S-P3 @ MOVE U+P2,S-P4 @ DRA
     W U-P2,S+P4
1115 I8=SQR(1.39/L9) @ T=T*I8 @
     Y1=Y1*I8 @ GOTO 1075 @ ! 1.
     39 IS 50% CH1-SQUARED VALUE
1120 MOVE K3-.225*S1,K6+.1*S2 @
     LABEL "50% & 95% CONFIDENCE
     ELLIPSES"
1125 X=LEN("MU="&B$)
1130 MOVE K3-.025*X*S1,K5-.2*S2
     @ LABEL "MU="&B$
1135 Y=LEN("SIGMA="&A$)
1140 MOVE K4+(.1-.05*(Y-1))*S1,K
     5-.2*S2 @ LABEL "SIGMA="&A$
1145 MOVE U+.6*S1,S-.56*S2 @ LAB
     EL "U" @ MOVE K3,K6+.02*S2
     @ LABEL "S" @ IF P8>=2 THEN
     COPY
1150 RETURN
1155 ! SUBROUTINE OF FOR SCALING
     THE N0.'S MU AND SIGMA ON
     GRAPH
1160 Z=ABS(Z1) @ IF FP(Z1)=0 AND
     ABS(Z1)<1000000000 THEN A$
     =VAL$(Z1) @ RETURN
1165 IF Z<1 THEN C$="-" ELSE C$=
     "+"
1170 T=LGT(Z) @ T1=FP(T)
1175 IF T1=0 THEN 1200 ELSE I8=I
     NT(T)
1180 T1=.5*(1-SGN(T))+T1 @ T1=10
     ^T1+.000005
1185 A$=VAL$(T1) @ Z=SGN(Z1)*VAL
     (A$[1,7]) @ IF FP(Z)=0 THEN
     1205
1190 A$=VAL$(Z) @ IF I8#0 THEN A
     $=A$&C$&VAL$(ABS(I8))
1195 RETURN
1200 A$=VAL$(SGN(Z1))&".0" @ A$=
     A$&C$&VAL$(IP(ABS(T))) @ RE
     TURN
1205 A$=VAL$(Z)&".0"&C$&VAL$(ABS
     (IP(I8))) @ RETURN
```

## REFERENCES


1. Brennan, L. E. and Reed, I. S., A Recursive Method of Computing the Q Function, IEEE TRANS. ON INFO. TH. April 1965, pp. 312-313.

2. Cody, W. J., Rational Chebyshev Approximations for the Error Function, MTAC (presently Math. Comp.), July 1969, pp. 631-637.

3. DiDonato, A. R. and Jarnagin, M. P., Integration of the General Bivariate Gaussian Distribution over an Offset Ellipse, NWL Report 1710, 11 Aug 1960, Naval Surface Weapons Center, Dahlgren, VA 22448.

4. DiDonato, A. R. and Jarnagin, M. P., Integration of the General Bivariate Gaussian Distribution over an Offset Circle, Math. Comp. 15, #76, Oct 1961, pp. 375-382.

5. DiDonato, A. R. and Jarnagin, M. P., A Method for Computing the Generalized Circular Error Function and the Circular Coverage Function, NWL Report No. 1768, 23 Jan 1962, Naval Surface Weapons Center, Dahlgren, VA 22448.

6. DiDonato, A. R. and Jarnagin, M. P., A Method for Computing the Circular Coverage Function, Math. Comp. 16, #79, July 1962, pp. 347-355.

7. DiDonato, A. R. and Jarnagin, M. P., Jr., Use of the Maximum Likelihood Method Under Quantal Responses for Estimating the Parameters of a Normal Distribution and Its Application to an Armor Penetration Problem, NWL Technical Report TR-2846, Nov 1972, Naval Surface Weapons Center, Dahlgren, VA 22448.

8. DiDonato, A. R. and Jarnagin, M. P., Jr., Maximum Likelihood Estimation in Quantal Response Experiments, SIAM J. Appl. Math. 26, #2, Mar 1974, pp. 447-454.

9. DiDonato, A. R., Jarnagin, M. P. and Hageman, R. K., Computation of the Bivariate Normal Distribution over Convex Polygons, NSWC/DL TR-3886, Sep 1978, Naval Surface Weapons Center, Dahlgren, VA 22448.

10. DiDonato, A. R., Jarnagin, M. P., and Hageman, R. K., Computation of the Integral of the Bivariate Normal Distribution over Convex Polygons, SIAM J. Sci. Stat. Comput., 1, #2, June 1980, pp. 179-186.

11. DiDonato, A. R., and Hageman, R. K., Computation of the Integral of the Bivariate Normal Distribution over Arbitrary Polygons, NSWC TR 80-166, June 1980, Naval Surface Weapons Center, Dahlgren, VA 22448.

12. DiDonato, A. R. and Hageman, R. K., A Method for Computing the Integral of the Bivariate Normal Distribution over an Arbitrary Polygon, SIAM J. Sci. Stat. Comput. 3, #4, Dec 1982, pp. 434-446.

13. Garland, K., Protective Ballistic Limits by Maximum Likelihood Method Adapted for Tektronix 4050 Series, NSWC TR 81-352, Sep 1981, Naval Surface Weapons Center, Dahlgren, VA 22448.

14. Morris, A. H., NSWC/DL Library of Mathematics Subroutines, NSWC TR 81-410, Oct 1981, Dahlgren, VA 22448.

15. Weingarten, H. and DiDonato, A. R., A Table of Generalized Circular Error, Math. Comp. 15, #74, April 1961, pp. 169-173.

## DISTRIBUTION LIST

Chief of Naval Operations
Attn: OP-982
Department of the Navy
Washington, D.C. 20350

Chief of Naval Operations
Attn: OP-982E
Department of the Navy
Washington, D.C. 20350

Chief of Naval Operations
Attn: OP-98[illegible]
Department of the Navy
Washington, D.C. 20350

Chief of Naval Operations
Attn: OP-987
Department of the Navy
Washington, D.C. 20350

Chief of Naval Operations
Attn: OP-961
Department of the Navy
Washington, D.C. 20350

Commander, Naval Facilities Engineering
Command
Department of the Navy
Attn: Code 032F
200 Stovall Street
Alexandria VA 22332

Office of Naval Research
Code 411
800 North Quincy Street
Arlington, VA 22217

Commander, Naval Sea Systems Command
Attn: SEA 05R
Department of the Navy
Washington, D.C. 20362

Fleet Analysis Center
Naval Weapons Station
Seal Beach
Attn: Library
Corona, CA 91720

Commander, U.S. Naval Weapons Center
Attn: Code 38
China Lake, CA 93555

U.S. Naval Observatory
34th Street and Massachusetts Avenue, N.W.
Attn: Library
Washington, D.C. 20390

U.S. Naval Oceanographic Office
Navy Lib./NSTL Station
Bay St. Louis, MS 39522

The Library of Congress
Attn: Exchange and Gift Division
Washington, D.C. 20540

Director
Office of Naval Research Branch Office
1030 East Green Street
Pasadena, CA 91101

Commanding Officer
Marine Aviation Detachment
Pacific Missile Test Center
Point Mugu, CA 93042

Director
U.S. Army Ballistic Research Laboratory
Attn: DRDAR-TSB-S (STINFO)
Aberdeen Proving Ground, MD 21005

Director
National Security Agency
Attn: Library
Fort George G. Meade, MD 20755

Commanding General
White Sands Missile Range
Attn: Technical Library, Documents Section
White Sands Missile Range, NM 88002

Director
National Bureau of Standards
Attn: Library
Gaithersburg, MD 20760

Argonne National Laboratory
Attn: Dr. A.H. Jaffey, Bldg. 200
9700 South Cass Avenue
Argonne, IL 60439

Rand Corporation
Attn: Library D
1700 Main Street
Santa Monica, CA 90406

University of Chicago
Attn: Prof. W. Kruskal, Statistics Department
5734 Univ. Ave.
Chicago, IL 60637

Joint Strategic Target Planning Staff
OFFUTT Air Force Base, NB 68113

Massachusetts Institute of Technology
Attn: Computation Center
Cambridge, MA 02139

President
Naval War College
Newport, RI 02840

Dr. George Ioup, Physics Department
University of New Orleans
Lake Front, New Orleans, LA 70148

William C. Guenther
Box 3332 University Station
Laramie, WY 82071

Dr. Donald E. Amos, 5642
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185

Donald Baker Moore
Exploratory Technology
P.O. Box KK
Fairfield, CA 94533

Dr. I. Sugai, F4A
The Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20707

Prof. George Marsaglia
Computer Science Department
Washington State University
Pullman, WA 99164

Prof. H. Goldstein
Columbia University
520 W. 120th Street, Engter Bldg.
New York, NY 10027

Prof. J. Gurland
University of Wisconsin-Madison
Department of Statistics
1210 West Dayton Street
Madison, WI 53706

John A. Simpson
Chief of Missile and Projectile Systems Design
Norden Systems, Inc.
P.O. Box 5300
Norwalk CT 06856

M. Miller
Shering Plough Corp.
60 Orange Street
Bloomfield, NJ 07003

Prof. Joo Koo
Department of Mathematics
Northeastern Illinois University
Chicago, ILL 60625

Harvey S. Picker
Physics Department
Trinity College
Hartford, CT 06106

Naval Ocean Systems Center
Attn: Library
San Diego, CA 92152

H. Saunders
Building 41. Room 304
General Electric Company
One River Road
Schenectady, NY 12345

R. F. Hausman
Lockheed Missile & Space Company
Department 6212 – Bldg. 104
P.O. Box 504
Sunnyvale, CA 94088

Dr. Harold Crutcher
35 Westall Avenue
Asheville, NC 28804

Deputy Director Marine Corps
Operations Analysis Group
Attn: Dr. B. Barfoot
2000 N. Beauregard Street
Alexandria, VA 22311

Dr. H. Weingarten
LMSC
O/8440 B 538
P.O. Box 504
Sunnyvale, CA 94088-3504

Prof. D. C. Hoaglin
Dept. of Statistics
Harvard University
1 Oxford Street
Cambridge, MA 02138

California Institute of Technology
Attn: Prof. T. Y. Wu
Pasadena, CA 91109

RTS-2A1
Defense Intelligence Agency
Washington, D.C. 20301

Naval Ship Research & Development Center
Attn: Code 1120 (E. O'Neill)
Code 1129 (E. Wolfe)
Code 1127 (S. Baylin)
Bethesda, MD 20084

Office of Naval Research
Attn: Math & Sciences Division
Library
Washington, D.C. 20360

Commander
David W. Taylor Naval Ship R&D Center
Attn: Technical Library, code 5220
Bethesda, Maryland 20084

Commander, David W. Taylor
Naval Ship Research and Dev. Center
Attn: Code 1805
Bethesda, MD 20034

Commander, David W. Taylor
Naval Ship Research and Development Center
Attn: Code 184
Bethesda, MD 20034

George Cyrcos
AVARDA
DAVAA-N-SA
Fort Monmouth, NJ 07703

Peter Shugart
U.S. Army Tradoc Systems Analysis Activity
Attn: ATAA-TDD
White Sands Missile Range, NM 88002

Director, Naval Research Laboratory
Attn: Library
Washington, D.C. 20390

Local Distribution

D
D1
E
E411
F14 (E. Morgan)
F26 (R. Muliken)
F42 (E. Spooner)
G
G12 (Dr. R. Hinkle)
G12 (G. Hornbaker)
G102 (F. Clodius)
G301 (G. Seidl)
K
K04
K10
K14 (D. Snyder)
K33 (A. Morris)
K40
K104 D (30)
K105 (V. Patton)
K106 (Dr. M. Thomas) (3)
K106 (Dr. J. Crigler)
K106 (G. Gemmill)
N
N31 (S. Vittoria)
R44
R44 (Dr. A.H. Van Tuyl)
E431 (6)
E432 (2)

END

FILMED

9-85

DTIC