THE ARIZONA STATE UNIVERSITY NEAREST NEIGHBOR PROGRAM: DOCUMENTATION AND DISCUSSION

Joel C. Johnstone
Richard W. Effland
Geoffrey A. Clark

Department of Anthropology
Arizona State University
Tempe, Arizona 85281 U.S.A.

A program is described which generates data point, nearest neighbor and significance test statistics for use with large samples (up to 1,000 points) in one to n-way nearest neighbor analyses. The program can be interfaced with a graphic plotting routine (GIPSY) for visual display and/or comparison of point and shared area distributions (cf. Clark, Effland and Johnstone, this volume).

In a series of recent papers, Whallon (1973,1974) has presented and discussed multivariate statistical procedures appropriate to the evaluation of the significance of artefact clusters on the surfaces of archaeological sites. Amongst the most promising of these are a series of techniques grouped under the rubric of "nearest neighbor analysis" (Clark and Evans 1954; Pielou 1959,1969; Thompson 1956; Pinder and Witherick 1972; Dacey 1963). Although there are different kinds of nearest neighbor analyses (e.g. lineal, areal), the basic approach was designed to provide the investigator with an objective measure of the degree of departure from randomness toward maximal dispersion or aggregation of point scatters on two-dimensional plane surfaces. From an archaeological perspective, these point scatters could correspond to the co-ordinate locations of various artifact and faunal debris categories scattered across site surfaces; they might also represent the locations of sites within regions or macroenvironmental zones, or architectural features or units within sites.

It is not our intent here to provide a critique of nearest neighbor analysis. The approach itself, and a battery of statistics which can be used to advantage in conjunction with it are discussed by Whallon (1974) and by Clark, Effland and Johnstone (1977) elsewhere in this volume. Rather we outline here a FORTRAN program which is designed to facilitate nearest neighbor analysis when the number of data points is large. Some preliminary remarks, however, seem advisable lest fundamental misconceptions arise at the outset about the uses and limitations of the approach.

First, it should be noted that the statistic is highly sensitive to area. Caution is advised in the determination of area, especially if site boundaries are arbitrary and/or if not sampling design has been incorporated into data collection. Second, although the method is not constrained by the size and shape criteria which limit the usefulness of dimensional analysis

of variance (Whallon 1973), nearest neighbor analysis does require that the data be point provenienced (i.e. recorded as a series of two (or three) dimensional rectangular co-ordinates). Normally, first order nearest neighbor distances are used to compile the descriptive statistics which in turn provide a basis for tests of significance and of association. Second, third, fourth . . . nth order nearest neighbors can also be used, however, for more sophisticated kinds of analyses in which data are organized into hierarchies (e.g. central place studies).

If samples are large, manual calculation of even first order nearest neighbor statistics can be tedious and time consuming. For this reason, the following FORTRAN program has been developed. It 1) yields the series of basic statistics described below, and 2) can be interfaced with a graphic plotting program (GIPSY4)(Monmonier 1968) designed to plot from a 30" CALCOMP drum plotter. The version of GIPSY nowin use was modified by F. Aldrich (Geography) and R. Effland (Anthropology) for the UNIVAC 1110 system currently in use at Arizona State University. These plotting routines provide a versatile package of mapping options, including line, symbol, contour and text plotting. Symbols and "cut-off" circles (Whallon 1974:22,23; Hanson 1975) are automatically formatted and scaled by the nearest neighbor program, and an exterior border is established which corresponds to user-specified dimensions (i.e. it represents graphically the area to be used in the analysis, although it may not correspond to it exactly in shape). Additional line, contour and/or text can be added as desired. The output data are stored in temporary disc file space until entered into the plotting runstream. The main program given below, interfaced with a CALCOMP plotter, produced the graphic output shown in Figs. 7-14 in Clark, Effland and Johnstone (cf. this volume).

```
C          PROGRAM ID:  NEAREST NEIGHBOR PROGRAM
C**********IDENTIFICATION SECTION
C          PROGRAMMER:  J.C. JOHNSTONE
C          DATE WRITTEN:  10/10/1975
C          DATE COMPILED:  10/17/1975
C          LANGUAGE:  FORTRAN V
C          INSTALLATION:  ASU - UNIVAC-1100
C
C          THE ENTIRE PROGRAM IS DIVIDED INTO THREE PORTIONS:
C              1) NEIGHBOR - THE MAIN PROGRAM IN WHICH PARAMETERS AND
C                 AND OPTIONS ARE SPECIFIED AND CHECKED.
C              2) ANEAR - SUBPROGRAM IN WHICH DATA OF PRE-SPECIFIED
C                 PARAMETERS ARE READ IN AND ALL CALCULATIONS ARE DONE.
C                 OPTIONAL TABLE GENERATION. PLOT OF DATA. HISTOGRAMS.
C              3) PLOT - INTERFACING SUBPROGRAM THAT CALLS TO OUTSIDE
C                 PLOTTING ROUTINES SPECIFIED BY OPTIONS AND DATA CARDS
C                 AT END OF MAIN DATA DECK.
C
C          NEIGHBOR - MAIN PROGRAM
C                VARIABLE DEFINITIONS
C   NAME      TYPE      FORM                DEFINITION
C   A          D        INP       AREA OF POINTS PLUS BUFFER IN SQUARE MEASURE
C   ANEAR      D        FUN       NAME OF FUNCTION SUBPROGRAM
C   C          I        INP       NUMBER OF CASES - PAIRS OF X,Y COORDINATES
C   D          D        INT       ARRAY FOR NEIGHBOR DISTANCES
C   DN         D        INT       ARRAY FOR POINT DISTANCES AND POINT IDENTIFIER
C   M          I        INP       PLOTTING OPTION, 0=NONE, 1=YES
C   O1         I        INP       NEIGHBOR UPON WHICH STATISTICS ARE TO DONE,<=C
C   O2         I        INP       OPTION FOR TABLE OF FIRST FIVE NEIGHBORS
C   P          R        INT       ARRAY FOR STORAGE OF POINT COORDINATES
C   X          R        INT       DUMMY VARIABLE
```

```
C               THIS PORTION OF THE PROGRAM INITIALIZES ARRAYS AND READS IN:
C           THE NUMBER OF CASES, THE AREA, THE MAPPING OPTION, THE NUMBER OF
C           THE NEIGHBOR UPON WHICH STATISTICS ARE TO BE CALCULATED, AND
C           THE OPTION FOR THE TABLE OF THE FIRST FIVE NEIGHBORS FOR EACH POINT.
C           THE NUMBER OF CASES IS CHECKED TO SEE IF IT IS WITHIN THE DATA LIMITS
C           (LESS THAN OR EQUAL TO 1000) AND THE NUMBER OF THE NEIGHBOR UPON
C           WHICH STATISTICS ARE TO BE DONE IS CHECKED TO MAKE SURE IT IS WITHIN
C           THE ACTUAL NUMBER OF CASES. THEN IF ALL CHECKS OUT A CALL IS PLACED
C           TO THE SUBPROGRAM - ANEAR.
C
C***********PROGRAM SECTION
      INTEGER C,M/0/,B1/1/,B2/0/,UNIT/5/
      REAL ACARD,IOTEST,INDATA
      REAL*8 A,ANEAR,D,DN,P,X
      DIMENSION P(3,1000),D(1000),DN(2,1000),IOTEST(8),INDATA(10)
      DATA IOTEST/5HINPUT,4HDATA,5HCASES,5HLIMIT,4HUNIT,4HAREA,5HTABLE,
     *4HPLOT/
      READ(5,102,END=20) ACARD
      IF (ACARD .NE. IOTEST(1)) GO TO 11
   10 READ(5,103,END=20) ACARD,(INDATA(I),I=1,10)
      IF (ACARD .EQ. IOTEST(2)) GO TO 18
      IF (ACARD .EQ. IOTEST(3)) GO TO 12
      IF (ACARD .EQ. IOTEST(4)) GO TO 13
      IF (ACARD .EQ. IOTEST(5)) GO TO 14
      IF (ACARD .EQ. IOTEST(6)) GO TO 15
      IF (ACARD .EQ. IOTEST(7)) GO TO 16
      IF (ACARD .EQ. IOTEST(8)) GO TO 17
   11 WRITE(6,104) ACARD
      GO TO 19
   12 C=INTG(INDATA,10)
      GO TO 10
   13 B1=INTG(INDATA,10)
      GO TO 10
   14 UNIT=INTG(INDA  ,10)
      GO TO 10
   15 A=REA(INDATA,10)
      GO TO 10
   16 B2=IOL(INDATA,10)
      GO TO 10
   17 M=INTG(INDATA,10)
      GO TO 10
C  *********CHECK INPUT FOR CORRECTNESS.
   18 IF (C .GT. 1000) GO TO 19
      IF (B1 .GT. C) GO TO 19
      IF (A .EQ. 0.0) GO TO 19
C  *********DIVERT CONTROL TO SUBPROGRAM ANEAR.
      X=ANEAR(P,C,D,DN,A,M,B1,B2,UNIT)
       GO TO 20
   19 WRITE(6,100)
  20  WRITE(6,101)
C  *********FORMAT STATEMENT SECTION
  100 FORMAT(' ',56X,'DATA LIMITS EXCEEDED')
  101 FORMAT(' ',56X,'END OF PROGRAM (MAIN)')
 102  FORMAT(A6)
 103  FORMAT(A6,9X,10A1)
 104  FORMAT(' ','INCORRECT INPUT COMMAND - ',A6)
      CALL EXIT
      END


      FUNCTION ANEAR(POINT,NCASE,DIST,DAN,AREA,MAP,NOPT1,NOPT2,UNIT)
C
C***********IDENTIFICATION SECTION
C             PROGRAM ID:  SUBPROGRAM ANEAR
C             PROGRAMMER:  J.C. JOHNSTONE
C             LANGUAGE:  FORTRAN V
C             INSTALLATION:  ASU - UNIVAC-1100
C
C             ANEAR - SUBPROGRAM
C                     VARIABLE DEFINITIONS
C      NAME        TYPE        FORM                      DEFINITION
C     A             D           INT        3.1415927*DEN
C     AREA          D           INP        PASSED FROM MAIN. AREA OF POINTS
C     CASE          D           INT        REAL*8 FORM OF NCASE
C     CLL           D           INT        LOWER CONFIDENCE LEVEL=SQRT(2*DF-1)-1.96)**2/DF
C     CLU           D           INT        UPPER CONFIDENCE LEVEL=SQRT(2*DF-1)+1.96)**2/DF
C     CR1           D           INT        CUTOFF RADIUS=MEAN+1*XSIG
C     CR2           D           INT        CUTOFF RADIUS=MEAN+1.65*XSIG
C     CSQ           D           INT        CHI-SQUARE=2*A*XSQR
C     CSQA          D           INT        NORMAL APPROXIMATION=CSQ/CASE
C     D1            D           INT        SQUARE OF DIFFERENCES OF X COORDINATES
C     D2            D           INT        SQUARE OF DIFFERENCES OF Y COORDINATES
C     DAN           D           INT        ARRAY OF DISTANCES BETWEEN POINTS
C     DEN           D           INT        DENSITY=AREA/CASE
C     DF            R           INT        DEGREES OF FREEDOM=2*NCASE
C     DIST          D           INT        ARRAY FOR NEIGHBOR DISTANCES
C     DMAX          R           INT        MAXIMUM NEIGHBOR DISTANCE
C     DMIN          R           INT        MINIMUM NEIGHBOR DISTANCE
C     DNNS          D           INT        NEAREST NEIGHBOR STATISTIC=MEAN/DEXP
C     IAST          I           INT        ASTERISK SYMBOL NAME FOR HISTOGRAM
C     K             I           INT        PARAMETER OF SORT ROUTINE
C     KMN           I           INT        SORT PARAMETER=NCM1-1
C     L             I           INT        COUNT FOR DAN ARRAY
C     LINE          I           INT        LINE ARRAY FOR HISTOGRAM
C     MAP           I           INP        PLOTTING OPTION, PASSED FROM MAIN
C     MEAN          R           INT        MEAN OF NEIGHBOR DISTANCES=SUM/CASE
C     NCASE         I           INP        PASSED FROM MAIN. NUMBER OF CASES
C     NCM1          I           INP        PASSED FROM MAIN. =NCASE-1
C     NOPT1         I           INP        PASSED FROM MAIN. NEIGHBOR STATISTIC OPTION
C     NOPT2         I           INP        PASSED FROM MAIN. TABLE OPTION
C     NP            I           INT        FLAG FOR START OF TABLE
C     NPGE          I           INT        COUNT OF LINES ON A PAGE
C     NUM           I           INT        SORT PARAMETER
C     NUMM1         I           INT        SORT PARAMETER
C     POINT         D           INP        ARRAY FOR X,Y COORDINATES, 1=X,2=Y,3=*
C     RANGE         R           INT        RANGE OF NEIGHBOR DISTANCES
C     SAVE          D           INT        SAVE AERA FOR SORT ROUTINE
```

```
C   SC          D       INT     STANDARD NORMAL VARIATE 'Z'=(MEAN-DEXP)/SDR
C   SCALE       R       INT     SCALING FACTOR FOR HISTOGRAM=RANGE/10
C   SDR         D       INT     STANDARD ERROR=.26136/SQRT(CASE-DEN)
C   START       R       INT     STARTING POINT FOR HISTOGRAM
C   STDV        D       INT     NORMAL STANDARD VARIATE=SQRT(2*CSQ)-SQRT(2*DF-1)
C   SUM         R       INT      SUM OF NEIGHBOR DISTANCES
C   VALUE       R       INT     ARRAY FOR HISTOGRAM COLUMN VALUES
C   XMAX        R       INT      MAXIMUM X INPUT VALUE
C   XMEAN       R       INT      MEAN OF X INPUT VALUES
C   XMID        R       INT      MIDPOINT OF X INPUT VALUES
C   XMIN        R       INT      MINIMUM X INPUT VALUE
C   XRANGE      R       INT      RANGE OF X INPUT VALUES
C   XSIG        D       INT      STANDARD DEVIATION=SQRT(XVAR)
C   XSQR        R       INT      SUM OF THE SQUARES OF THE DISTANCES
C   XSUM        R       INT      SUM OF X INPUT VALUES
C   XVAR        D       INT      VARIANCE OF DISTANCES=(XSQR/CASE)-MEAN**2
C   YMAX        R       INT      MAXIMUM Y INPUT VALUE
C   YMEAN       R       INT      MEAN OF Y INPUT VALUES
C   YMID        R       INT      MIDPOINT OF Y INPUT VALUES
C   YMIN        R       INT      MINIMUM Y INPUT VALUE
C   YSUM E      R       INT      SUM OF Y INPUT VALUES
C   YRANGE      R       INT      RANGE OF Y INPUT VALUES
C
C              THIS PORTION OF THE PROGRAM CALCULATES INPUT POINT STATISTICS.
C          DISTANCES BETWEEN ANY POINT AND ALL OF ITS NEIGHBORS, SORTS THESE BY
C          DISTANCE THEN USING OPTION ONE (NOPT1) PICKS OUT THE NEIGHBOR UPON
C          WHICH THE STATISTICS ARE TO BE CALCULATED.  USING THSE DISTANCES
C          IT CALCULATES NEIGHBOR STATISTICS AND PRINTS A SCALED HISTOGRAM.
C          FINALLY IF THE PLOTTING OPTION (MAP) IS SPECIFIED IT PLACES A CALL
C          TO THE SUBPROGRAM - PLOT.
C


C**********PROGRAM SECTION
      INTEGER FLAG,LINE,IAST,K,KNN,L,MAP,NCASE,NCM1,NOPT1,NOPT2,NP,
     1        NPOE,NUM,NUMM1,BLANK,UNIT
      REAL*8  A,AREA,CASE,CLL,CLU,CR1,CR2,CSQ,CSQR,D1,D2,DAN,DEN,DEXP,
     1        DIST,DNNS,POINT,SAVE,SC,SDR,STDV,XSIG,XVAR,FAC
      REAL    DF,DMAX,DMIN,MEAN,RANGE,SCALE,START,SUM,VALUE,XMAX,XMEAN,
     1        XMID,XMIN,XRANGE,XSQR,XSUM,YMAX,YMEAN,YMID,YMIN,YRANGE,
     2        YSUM
      DIMENSION DAN(2,NCASE),DIST(NCASE),LINE(10),POINT(3,NCASE),
     1          VALUE(2,10)
      DATA XMAX/0.0/,XMIN/99999.9/,XSUM/0.0/,YMAX/0.0/,YMIN/99999.9/,
     1     YSUM/0.0/,NP/1/,NPOE/60/,FLAG/0/,IAST/1H*/,LINE/10*1H /,
     2     BLANK/1H /

C
C  ********READ IN X, Y COORDINATE POINTS.
      NCM1=NCASE-1
      DO 2 I=1,NCASE
      READ(UNIT,1001) POINT(1,I),POINT(2,I)
        POINT(3,I)=I
        IF (NPOE .LT. 55) GO TO 1
          WRITE(6,1010)
          WRITE(6,1000)
          NPOE=3
    1   WRITE(6,1002) POINT(3,I),POINT(1,I),POINT(2,I)
        NPOE=NPOE+1
C
C  ********CALCULATE POINT STATISTICS.
        XSUM=XSUM+POINT(1,I)
        YSUM=YSUM+POINT(2,I)
        IF (POINT(1,I) .GT. XMAX)XMAX=POINT(1,I)
        IF (POINT(1,I) .LT. XMIN)XMIN=POINT(1,I)
        IF (POINT(2,I) .GT. YMAX)YMAX=POINT(2,I)
        IF (POINT(2,I) .LT. YMIN)YMIN=POINT(2,I)
        DIST(I)=0.0
    2 CONTINUE
C  ********CALCULATE POINT STATISTICS CONTINUED.
      XRANGE=XMAX-XMIN
      YRANGE=YMAX-YMIN
      XMID=(XMAX+XMIN)/2
      YMID=(YMAX+YMIN)/2
      XMEAN=XSUM/NCASE
      YMEAN=YSUM/NCASE
C
      IF (NPOE .LE. 42)GO TO 3
        WRITE(6,1010)
        NPOE=1
    3 WRITE(6,1003) XMEAN,YMEAN,XRANGE,YRANGE,XMAX,XMIN,YMAX,YMIN,XMID,
     1 YMID
      NPOE=NPOE+13
C
C  ********CALCULATION OF POINT TO POINT DISTANCES.
      NNOR=NOPT1
      DO 23 II=1,NNOR
      NOPT1=II
      XSQR=0.0
      SUM=0.0
      DMAX=0.0
      DMIN=99999.9
      DO 100 IFL=1,20
  100 VALUE(IFL)=0.0
      DO 14 I=1,NCASE
        L=1
        DO 4 J=1,NCASE
          IF (I .EQ. J)GO TO 4
C  ********CALCULATION OF POINT DISTANCE.
            D1=(POINT(1,I)-POINT(1,J))*(POINT(1,I)-POINT(1,J))
            D2=(POINT(2,I)-POINT(2,J))*(POINT(2,I)-POINT(2,J))
            DAN(1,L)=DSQRT(D1+D2)
            DAN(2,L)=POINT(3,J)
            L=L+1
```

```
    4    CONTINUE
C
C   ********CALCULATION OF MAXIMUM SORT PARAMETER.
        IF(NOPT2 .EQ. 0)K=NOPT1
        IF((NOPT2 .NE. 0) .AND. (NOPT1 .LE. 5))K=5
        IF((NOPT2 .NE. 0) .AND. (NOPT1 .GT. 5))K=NOPT1
C
C   ********SORT ROUTINE
        DO 7 M=1,K
          KMN=NCM1-M
          FLAG=0
          DO 6 N=1,KMN
          NUM=(NCM1+1)-N
          NUMM1=NUM-1
          IF (DAN(1,NUM) .GE. DAN(1,NUMM1))GO TO 6
              FLAG=1
              DO 5 NN=1,2
              SAVE=DAN(NN,NUM)
              DAN(NN,NUM)=DAN(NN,NUMM1)
              DAN(NN,NUMM1)=SAVE

    5         CONTINUE
    6       CONTINUE
            IF (FLAG .EQ. 0)GO TO 8
    7     CONTINUE
C
C
    8     IF (NOPT2 .EQ. 0)GO TO 13
C   ********PRINT-OUT OF FIRST FIVE NEIGHBOR TABLE.
            IF (NP .EQ. 0)GO TO 9
              NP=0
              IF (NPGE .GE. 40)GO TO 10
                GO TO 11
    9         IF (NPGE .LT. 55)GO TO 12
   10           WRITE(6,1010)
                NPGE=1
   11             WRITE(6,1004)
                  NPGE=NPGE+4
   12         WRITE(6,1005) POINT(3,I),DAN(2,1),DAN(1,1),DAN(2,2),DAN(1,2),
     1 DAN(2,3),DAN(1,3),DAN(2,4),DAN(1,4),DAN(2,5),DAN(1,5)
              NPGE=NPGE+1
C   ********SAVE DISTANCE OF DESIRED NEIGHBOR.
   13     DIST(I)=DAN(1,NOPT1)
C
C   ********CALCULATE NEIGHBOR STATISTICS.
          XSQR=XSQR+DIST(I)*DIST(I)
          SUM=SUM+DIST(I)
          IF (DIST(I) .GT. DMAX)DMAX=DIST(I)
          IF (DIST(I) .LT. DMIN)DMIN=DIST(I)
   14   CONTINUE
C   ********CALCULATE NEIGHBOR STATISTICS CONTINUED.
        CASE=NCASE
        DEN=CASE/AREA
        MEAN=SUM/CASE
        DEXP=(1./(DSQRT(DEN)))*(((FAC(2*NOPT1))*NOPT1)/(((2**NOPT1)*(FAC(*
       *OPT1)))**2))
        XVAR=(XSQR/CASE)-MEAN**2
        XSIG=SQRT(XVAR)
        DNNS=MEAN/DEXP
        SOR=0.26136/DSQRT(CASE*DEN)
        SC=(MEAN-DEXP)/SOR
        A=3.1415927*DEN
        CSQ=2.0*A*XSQR
        DF=2*NCASE*NOPT1
        STDV=SQRT(2.0*CSQ)-SQRT(2.0*DF-1.0)
        CSQA=CSQ/CASE
        CLL=(SQRT(2.*DF-NOPT1)-1.)-1.96)**2/DF
        CLU=(SQRT(2.*DF-NOPT1)-1.)+1.96)**2/DF
        CR1=MEAN+1.0*XSIG
        CR2=MEAN+1.65*XSIG
C
        IF (NPGE .LT. 28)GO TO 15
          WRITE(6,1010)
          NPGE=1
   15   WRITE(6,1006) NOPT1
        WRITE(6,1007) NCASE,AREA,DEN,DEXP,SUM,MEAN,DNNS,XVAR,XSIG,SOR,SC,
       * CSQ,DF,STDV,CSQA,CLL,CLU,CR1,CR2
        WRITE(6,1014)
C
C   ********HISTOGRAM ROUTINE
        RANGE=DMAX-DMIN
C   ********CALCULATE SCALING FACTOR.
        SCALE=RANGE/10
        NCOUNT=NCASE
        DO 17 I=1,NCASE
          START=DMIN
          DO 16 K=1,10
            START=START+SCALE
            VALUE(2,K)=START
            IF (DIST(I) .GT. START)GO TO 16
              VALUE(1,K)=VALUE(1,K)+1
              NCOUNT=NCOUNT-1
              GO TO 17
   16     CONTINUE
   17   CONTINUE
        FLAG=0
        VALUE(1,10)=VALUE(1,10)+NCOUNT
        DO 20 I=1,NCASE
          J=(NCASE+1)-I
          DO 18 K=1,10
            IF (VALUE(1,K) .LT. J)GO TO 18
              LINE(K)=IAST
              FLAG=1
```

```
  18     CONTINUE
         IF (FLAG .EQ. 0)GO TO 20
           IF (MOD(J,5) .NE. 0)GO TO 19
             WRITE(6,1011) J,((LINE(K),L=1,5),K=1,10)
             GO TO 20
  19     WRITE(6,1012)((LINE(K),L=1,5),K=1,10)
  20   CONTINUE
       WRITE(6,1013)(VALUE(2,I),I=1,10)
       NPGE=60

       IF (MAP .EQ. 0)GO TO 21
       IF (MAP .NE. 1) GO TO 21
C  ********MAP OPTION SPECIFIED - PASS CONTROL TO PLOT SUBPROGRAM.
       WRITE(6,1008)
C.....CALL TO PLOTTING PACKAGE
       Z=PLOT(NCASE,CR1,CR2,POINT,YMAX)
       MAP=0
  21   DO 22 K=1,20
  22   LINE(K)=BLANK
       ANEAR=ONNS
       NOPT2=0
  23   CONTINUE
       WRITE(6,1009)

C
C  ********FORMAT STATEMENT SECTION.
 1000 FORMAT(' ',56X,'INPUT DATA POINTS',/51X,'NUMBER',4X,'X CO-ORD',5X,
     1'Y CO-ORD')
 1001 FORMAT(2F10.5)
 1002 FORMAT(' ',45X,F10.0,4X,F10.5,3X,F10.5)
 1003 FORMAT('0',49X,'*****DATA POINT STATISTICS*****',//56X,'MEAN (X)..
     1....',F8.3,/56X,'MEAN (Y).....',F8.3,/56X,'RANGE (X).....',F8.3,/
     256X,'RANGE (Y).....',F8.3,/56X,'MAXIMUM (X)...',F8.3,/56X,'MINIMUM
     3 (X)...',F8.3,/56X,'MAXIMUM (Y)...',F8.3,/56X,'MINIMUM (Y)...',F8.
     43,/56X,'MID-POINT (X).',F8.3,/56X,'MID-POINT (Y).',F8.3)
 1004 FORMAT('0',50X,'*****NEAREST NEIGHBOR DISTANCES*****',/15X,'NUMBER
     1',3X,'POINT 1.....DIST',4X,'POINT 2.....DIST',4X,'POINT 3.....DIST
     2',4X,'POINT 4.....DIST',4X,'POINT 5.....DIST',/)
 1005 FORMAT(' ',9X,F10.0,5(F10.0,F10.3))
 1006 FORMAT('0',49X,'****NEAREST NEIGHBOR STATISTICS****',/55X,'OF NUMB
     1ER ',I4,' NEIGHBOR',/)
 1007 FORMAT(' ',43X,'1.PARAMETRIC',/46X,'NUMBER OF POINTS ''N''',11X,
     =I4,/46X,'AREA ''A''',7X,F10.3,/46X,'DENSITY',24X,F10.5,/46X,'EXPECTED D
     =ISTANCE ''RE''',9X,D15.8,/46X,'SUM OF DISTANCES',15X,F10.5,/46X,
     ='MEAN OF DISTANCES ''R0''',9X,F10.5,/46X,'NEAREST NEIGHBOR STATIST
     =IC ''R''',1X,F10.5,/46X,'VARIANCE OF DISTANCES',10X,F10.5,/46X,
     ='STANDARD DEVIATION OF DISTANCE ',F10.5,/46X,'STANDARD ERROR',17X,
     =F10.5,/46X,'STANDARD NORMAL VARIATE ''Z''',4X,D15.8,//44X,'11.CHI-
     =SQUARE (NON-PARAMETRIC)',/46X,'CHI-SQUARE',21X,F10.5,/46X,'DEGREES
     = OF FREEDOM',13X,F10.5,/46X,'NORMAL STANDARD VARIATE ',8X,F10.5,/
     =46X,'NORMAL APPROXIMATION',11X,F10.5,/46X,'5% CONFIDENCE LEVEL',6X
     =,F8.3,' TO ',F8.3,//44X,'III.CUT OFF RADII',/46X,'RADIUS (1=STD.DE
     =V*MEAN)    ',5X,F10.5,/46X,'RADIUS (1.65=STD.DEV*MEAN)    ',2X,F10.5
     =)
 1008 FORMAT('1',49X,'MAP OPTION SPECIFIED - DATA PLOTTED')
 1009 FORMAT('0',50X,'END OF SUBPROGRAM - END ANALYSIS')
 1010 FORMAT('1',' ')
 1011 FORMAT(' ',3X,I4,2H -,20(2X,3A1,1X))
 1012 FORMAT(' ',8X,1H!,20(2X,3A1,1X))
 1013 FORMAT(' ',8X,1H+,20(6H------),/10X,10(F6.2,6X))
 1014 FORMAT('1',//9X,1H+,38(1H-),35HNEAREST NEIGHBOR DISTANCE HISTOGRAM
     =,48(1H-),/9X,1H!)
      RETURN
      END


      FUNCTION PLOT(NCASE,CR1,CR2,POINT,YMAX)
C
C**********IDENTIFICATION SECTION
C           PROGRAM ID:  SUBPROGRAM PLOT
C           PROGRAMMER: R. EFFLAND, J.C. JOHNSTONE
C           LANGUAGE:  FORTRAN V
C           INSTALLATION:  ASU - UNIVAC-1100
C
C           PLOT - SUBPROGRAM
C
C**********PROGRAM SECTION
      INTEGER NCASE,IFLG
      REAL ITEST,ACARD,INDATA,YMAX
      REAL*8 CR1,CR2,POINT
      DIMENSION  POINT(3,NCASE),ITEST(7),INDATA(10)
      DATA ISYM8/2/,SIZE/1./,SCAL/0./,X80RD/.01/,Y80RD/.01/,ICR/1/.
     = ITEST/6HFINISH,6HSYM80L,4HSIZE,5HSCALE,6HX-80RD,6HY-80RD.
     = 6HOPTION/,IFLG/0/
C.....SCALE,SHIFT,80RDER, AND SYM80L PACKAGES WILL BE CREATED IN THIS
C.....ROUTINE.  SHIF WILL BE SET AT 1 SO WRITE ALL TEXT PRIOR TO THE
C.....ADDITION OF THIS ROUTINE INTO THE PLOTTING PROGRAM.
C.....CARD ORDER====ASSIGN UNIT 4.      (2)PLOT DATA CARD  (3)XQT OF GIPSY
C.....  (4)TEXT OR OTHER GISPY PACKAGES OPTIONAL  (5)ADD UNIT 4.  (6)FIN
       ISYM = 31
       NN = 1
       NSYMB = 2*NCASE
    5  READ(5,50) ACARD,(INDATA(I),I=1,10)
       IF (ACARD .EQ. ITEST(1)) GO TO 10
       IF (ACARD .EQ. ITEST(2)) ISYMB=INTO(INDATA,10)
       IF (ACARD .EQ. ITEST(3)) SIZE=REA(INDATA,10)
       IF (ACARD .NE. ITEST(4)) GO TO 7
         SCAL=REA(INDATA,10)
         IFLG=1
         GO TO 5
    7  IF (ACARD .EQ. ITEST(5)) X80RD=REA(INDATA,10)
       IF (ACARD .EQ. ITEST(6)) Y80RD=REA(INDATA,10)
       IF (ACARD .EQ. ITEST(7)) ICR=INTO(INDATA,10)
       GO TO 5
   10  IF (IFLG .NE. 0) GO TO 20
        SCAL=25./YMAX
        IF (ICR .EQ. 1) SCAL=25./(YMAX+(2.*CR1))
        IF (ICR .EQ. 2) SCAL=25./(YMAX+(2.*CR2))
       IF (SCAL .GT. 1.0) SCAL=1.0
```

```
   20  WRITE (*,200)SCAL
C.....IF ICR = 0 THEN ONLY THE DATA POINTS WILL BE PLOTTED
C.....IF ICR = 1 THEN A CIRCLE OF 1.0 STD. DEVIATIONS + MEAN DISTANCE WILL PLOT
C.....IF ICR = 2 THEN THE CIRCLE RADIUS WILL BE 1.65 STD. DEV. + THE MEAN DIST.
       IF ((ICR-1)25,30,35
    25 WRITE (4,300)NCASE
C.....WRITE SYMBOL PACKAGE FOR DATA POINTS ONLY
       DO 110  I=1,NCASE
       WRITE(4,100)POINT(1,I),POINT(2,I),ISYMB,SIZE
   110 CONTINUE
       GO TO 199
    30 WRITE(4,300)NSYMB
C.....WRITE SYMBOL PACKAGE FOR CIRCLES AND DATA POINTS
C.....CIRCLES HAVE A RADIUS SIZE = TO 1.0 STD. DEVIATION + MEAN DISTANCE
       DO 120  I=1,NCASE
       WRITE (4,100)POINT(1,I),POINT(2,I),ISYM,CR1
       WRITE (4,100)POINT(1,I),POINT(2,I),ISYMB,SIZE
   120 CONTINUE
       GO TO 199
    35 WRITE (4,300)NSYMB
C.....WRITE SYMBOL PACKAGE FOR CIRCLES AND DATA POINTS
C.....CIRCLES HAVE A RADIUS = TO 1.65 STD. DEVIATIONS + MEAN DISTANCE
       DO 150  I=1,NCASE
       WRITE (4,100)POINT(1,I),POINT(2,I),ISYM,CR2
       WRITE (4,100)POINT(1,I),POINT(2,I),ISYMB,SIZE
   150 CONTINUE
   199 WRITE(4,600)XBORD,YBORD,NN
       WRITE(4,400)NN
       WRITE(6,500)
C  ****FORMAT STATEMENT SECTION
    50   FORMAT(A6,9X,10A1)
   100  FORMAT (2F10.5,5X,I5,F10.5)
   200  FORMAT ('SCAL',1X,F10.5)
   300  FORMAT ('SYMB',1X,I5)
   400  FORMAT ('SHIF',1X,I5)
   500  FORMAT (57X,'END OF MAP PROCESSING',/,55X,
      *  'MAP DATA PLACED ON UNIT 4.',/////)
   600  FORMAT ('BORD',1X,2F10.5,I5)
        PLOT=1.0
        RETURN
        END



        DOUBLE PRECISION FUNCTION FAC(N)
        REAL*8 SUM
        SUM=1.0
        DO 100 I=1,N
        SUM=SUM*I
   100  CONTINUE
        FAC=SUM
        RETURN
        END



        INTEGER FUNCTION INTG(IDATA,N)
         INTEGER COUNT,BLANK
        DIMENSION IDATA(N)
        DATA BLANK/1H /
        INTG=0
        COUNT=0
        DO 10 I=1,N
          J=(N+1)-I
          IF (IDATA(J) .EQ. BLANK) GO TO 10
                INTG=INTG+(NUM(IDATA,J)*(10**COUNT))
                COUNT=COUNT+1
    10  CONTINUE
        RETURN
        END



        REAL FUNCTION REA(IDATA,N)
        INTEGER SAVE,BLANK,COUNT,DECIM
        DIMENSION IDATA(N)
        DATA BLANK/1H /, DECIM/1H./
        SAVE=0
        COUNT=1
        DO 10 I=1,N
           IF (IDATA(I) .NE. DECIM) GO TO 10
           SAVE=I
           GO TO 11
    10  CONTINUE
        SAVE=11
    11  REA=INTG(IDATA,(SAVE-1))
        IF (SAVE .EQ. 11) GO TO 13
        J=SAVE+1
        DO 12 I=J,N
           IF (IDATA(I) .EQ. BLANK) GO TO 12
           REA=REA+(FLOAT(NUM(IDATA,I))/FLOAT(10**COUNT))
           COUNT=COUNT+1
    12  CONTINUE
    13  RETURN
        END
```

```
      INTEGER FUNCTION IBL(IDATA,N)
      INTEGER YES,NO,BLANK
      DIMENSION IDATA(N)
      DATA YES/1HY/,NO/1HN/,BLANK/1H /
      DO 11 I=1,N
         IF (IDATA(I) .EQ. BLANK) GO TO 11
         IF (IDATA(I) .NE. YES) GO TO 10
            IBL=1
            GO TO 12
   10    IBL=0
         GO TO 12
   11 CONTINUE
   12 RETURN
      END



      INTEGER FUNCTION NUM(NO,N)
      INTEGER ONE,TWO,THREE,FOUR,FIVE,SIX,SEVEN,EIGHT,NINE,ZERO
      DIMENSION NO(N)
      DATA ONE,TWO,THREE,FOUR,FIVE,SIX,SEVEN,EIGHT,NINE,ZERO/1H1,1H2,
     *1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0/
    1 IF(NO(N) .NE. ONE) GO TO 2
      NUM=1
      GO TO 11
    2 IF(NO(N) .NE. TWO) GO TO 3
      NUM=2
      GO TO 11
    3 IF (NO(N) .NE. THREE) GO TO 4
      NUM=3
      GO TO 11
    4 IF (NO(N) .NE. FOUR) GO TO 5
      NUM=4
      GO TO 11
    5 IF (NO(N) .NE. FIVE) GO TO 6
      NUM=5
      GO TO 11
    6 IF(NO(N) .NE. SIX) GO TO 7
      NUM=6
      GO TO 11
    7 IF (NO(N) .NE. SEVEN) GO TO 8
      NUM=7
      GO TO 11
    8 IF (NO(N) .NE. EIGHT) GO TO 9
      NUM=8
      GO TO 11
    9 IF (NO(N) .NE. NINE) GO TO 10
      NUM=9
      GO TO 11
   10 NUM=0
   11 RETURN
      END
```

The main program was written orginally by T.P. Muller (Chicago), and subsequently modified by G.A. Clark, S. Raab, C. Waters, R. Effland and J.C. Johnstone (Arizona State). The October 1975 version is given above. The program lists data points input by X and Y co-ordinates, and generates the following *data point statistics:*

Mean (X)	Minimum (X)
Mean (Y)	Maximum (Y)
Range (X)	Minimum (Y)
Range (Y)	Mid-point (X)
Maximum (X)	Mid-point (Y)

For each point, a tabular listing of first through fifth order neighbors is provided, and the program generates the following *nearest neighbor statistics:*

PARAMETRIC:

- Number of points (N)
- Area
- Density
- Expected distance ($\bar{r}_e$)
- Sum of distances
- Mean of distances ($\bar{r}_o$)
- Nearest neighbor statistic (R)
- Variance of distances
- Standard deviation of distances
- Standard error
- Standard normal variable (z)

CHI-SQUARED (NON-PARAMETRIC):

- Chi-squared
- Degrees of freedom
- Standard normal variable (z)
- Normal approximation
- Confidence interval (a = .05)

CUT-OFF RADII:

- 1.00 standard deviations
- 1.65 standard deviations

Finally, a histogram of the nearest neighbor distances is provided.

To summarize, point-provenienced data are input (format is 2F10.5). The user specifies 1) the number of points to be input (C,NCASE), 2) the area to be used in calculation (A,AREA—area is used to obtain the expected nearest neighbor distance ($\bar{r}_e$) and is critical to the analysis), 3) options for graphic interface (M,MAP), 4) the nearest neighbor to be computed (01,NØPT1), and 5) whether tabular output for the first five nearest neighbor distances is desired (02,NØPT2). The mapping options have been designed for a specific graphic interface (GIPSY4); certain internal modifications would be required for use with a different system. Alternatively, the mapping options can be bypassed (Map Option = 0). At present, the maximum number of data points is 1.000.

REFERENCES CITED

Clark, P.J. and F.C. Evans
1954 – Distance to nearest neighbor as a measure of spatial relationships in populations. ECOLOGY, Vol. 35, pp. 445-453.

Dacey, M.F.
1963 – Order neighbor statistics for a class of random patterns in multidimensional space.ANNALS OF THE ASSOCIATION OF AMERICAN GEOGRAPHERS, No. 53, pp. 505-515.

Hanson, G.T.
1975 – The analysis of spatial distributions in archaeological contexts: the Brand Site. Paper read at the 40th Annual Meetings of the Society for American Archaeology (Dallas, Texas; May, 1975).

Monmonier, M.S.

1968 – Computer mapping with the digital increment plotter. PROFESSIONAL GEOGRAPHER, Vol. 20, pp. 408, 409

Pielou, E.C

1959 – The use of point-to-plant distances in the study of the pattern of plant populations. JOURNAL OF ECOLOGY, Vol. 47, pp. 607-613.

1969 – AN INTRODUCTION TO MATHEMATICAL ECOLOGY. Wiley Interscience Series, John Wiley and Sons, Inc. New York.

Pinder, D.A. and M.E. Witherick

1972 – The principles, practice and pitfalls of nearest neighbor analysis. GEOGRAPHY, Vol. 57, pp. 277-288.

Thompson, H.R.

1956 – Distribution of distance to nth neighbor in a population of randomly distributed individuals. ECOLOGY, Vol. 37, pp. 391-394.

Whallon, R.

1973 – Spatial analysis of occupation floors I: The application of dimensional analysis of variance. AMERICAN ANTIQUITY, Vol. 38, pp. 266-278.

1974 – Spatial analysis of occupation floors II: The application of nearest neighbor analysis. AMERICAN ANTIQUITY, Vol. 39, pp. 16-34.