PLOTTING HOME RANGE FOR WILDLIFE DATA

Hoke S. Hill, Clemson University
Lawrence W. Grimes, Clemson University
Herbert C. Krasner, Clemson University

ABSTRACT

A SAS program is given which determines from a set of points representing animal sightings or locations, the boundary points of the animal's home range, computes the area of this home range, and displays the home range plot. The determination of the boundary points, points which form the convex hull of the observations, and the computation of the enclosed area are accomplished within a macro (RANGE) using PROC MATRIX. Graphical representation of the observations and home range are produced using PROC GPLOT. Illustrative examples and program listings are presented.

INTRODUCTION

Wildlife biologists are frequently interested in describing animal location and movement patterns. Many studies are made in which an animal is caught and a radio transmitter is attached in order to monitor the animal's movements. By taking directional readings of the animal's location from two or more fixed reference points, approximate "map" location can be determined. Each of these locations can in turn be translated into (X,Y) coordinates relative to a chosen origin. From a sequence of such location points, the wildlife biologist is interested in determing the animal's approximate home range location and area.

In this paper we assume a data set of (X,Y) coordinate points for which we determine the boundary points forming the convex hull or Mohr's minimum home range [1]. Included is the computation of the enclosed area using a trapezoidal method.

COMPUTATIONAL PROCEDURES FOR BOUNDARY AND AREA

The macro RANGE (see Appendix 1) for computing the home range boundary and area consists almost entirely of PROC MATRIX statements, making extensive use of the iterative and subscripting facilities within that procedure. Within the macro the data is first sorted by X Y using PROC SORT. Boundary point determination starts by considering the minimum X,Y valued pair (left end of the sorted table) to be the current pair. The following steps are then repeated until back at the left (minimum X,Y) end of the table:

1. Choose the next (X,Y) pair to be included on the boundary by calculating the slope between the current pair and all other pairs to the end of the table. The pair with the largest slope is picked to be next.
2. The trapazoidal area under the line from current to next is computed and either added to or subtracted from the accumulated total area depending on the direction in the table (left to right or reverse).
3. Next becomes current.
4. When the right end (maximum X,Y valued pair) of the table is reached, the direction is reversed and the steps are continued.

Whether or not each observation falls on the boundary and, if on the boundary, the order in which the pair is chosen are stored in columns of the matrix corresponding to that observation. Once the area has been computed, it is also stored with each observation. To facilitate plotting, the matrix is sorted by order of placement on the boundary with the starting (minimum X,Y) pair occurring twice, first and last. Finally a SAS data set is created from the matrix using the OUTPUT statement.

HOME RANGE PLOT EXAMPLES

A typical SAS data set output from macro RANGE is given as Table 1. Notice that values of Y are given the name YP or YB respectively according to whether the value of the indicator variable BOUND is 0 (not a boundary point) or 1 (boundary point). The home range boundary for this data set, plotted along with the data points gives the plot shown in Figure 1. This figure is generated by:

```
PROC GPLOT; BY SEASON AREA;
PLOT YP*X YB*X / OVERLAY;
```

The use of AREA as a by variable is simply to get the area value printed out on the plot.

Another possible application would involve the plotting of the home ranges for each of several seasons for a given animal, each of several animals for a given time period, etc., superimposed on the same graph. Our second example illustrates the plotting of seasonal home ranges for a given animal. The construction of the appropriate SAS data set is accomplished through several calls to macro RANGE for each of the seasonal data subsets and a subsequent merging of the output data sets. This composite data set is shown in Table 2, with the resulting plot shown as Figure 2. This plot is given by:

```
PROC GPLOT; BY F-AREA SP-AREA SU-AREA
                W-AREA TOT-AREA;
PLOT Y*X = SEASON;      (see Appendix 2).
```

The use of the three-variable plot option generates a legend indicating which symbol represents which season. The merging of each range's area onto every observation for use as a by variable allows the printing of the respective areas on the plot.

REFERENCE

1. Mohr, C.O. 1947. Table of equivalent populations of North American small mammals. Am. Midl. Nat. 37(1):223-249.

TABLE 1: DATA SET FOR PLOTTING LOCATIONS AND BOUNDARIES

X	Y	BOUND	ENTRY	YP	YB	SEASON	AREA
2	2	0	0	2	.	FALL	33
2	3	0	0	3	.	FALL	33
3	2	0	0	2	.	FALL	33
3	3	0	0	3	.	FALL	33
3	5	0	0	5	.	FALL	33
4	2	0	0	2	.	FALL	33
4	4	0	0	4	.	FALL	33
5	3	0	0	3	.	FALL	33
5	4	0	0	4	.	FALL	33
6	4	0	0	4	.	FALL	33
7	5	0	0	5	.	FALL	33
8	4	0	0	4	.	FALL	33
8	5	0	0	5	.	FALL	33
1	2	1	1	.	2	FALL	33
2	5	1	2	.	5	FALL	33
5	6	1	3	.	6	FALL	33
9	6	1	4	.	6	FALL	33
11	5	1	5	.	5	FALL	33
9	3	1	6	.	3	FALL	33
5	1	1	7	.	1	FALL	33
1	2	1	8	.	2	FALL	33

TABLE 2: DATA SET FOR PLOTTING MULTIPLE HOME RANGES

X	Y	BOUND	ENTRY	AREA	SEASON	F_AREA	SP_AREA	SU_AREA	W_AREA	TOT_AREA
1	2	1	1	33.0	FALL	33	24	35	56	90.5
2	5	1	2	33.0	FALL	33	24	35	56	90.5
5	6	1	3	33.0	FALL	33	24	35	56	90.5
9	6	1	4	33.0	FALL	33	24	35	56	90.5
11	5	1	5	33.0	FALL	33	24	35	56	90.5
9	3	1	6	33.0	FALL	33	24	35	56	90.5
5	1	1	7	33.0	FALL	33	24	35	56	90.5
1	2	1	8	33.0	FALL	33	24	35	56	90.5
9	6	1	1	24.0	SPRING	33	24	35	56	90.5
12	9	1	2	24.0	SPRING	33	24	35	56	90.5
14	8	1	3	24.0	SPRING	33	24	35	56	90.5
16	6	1	4	24.0	SPRING	33	24	35	56	90.5
15	5	1	5	24.0	SPRING	33	24	35	56	90.5
12	3	1	6	24.0	SPRING	33	24	35	56	90.5
10	4	1	7	24.0	SPRING	33	24	35	56	90.5
9	6	1	8	24.0	SPRING	33	24	35	56	90.5
9	6	1	1	35.0	SUMMER	33	24	35	56	90.5
10	8	1	2	35.0	SUMMER	33	24	35	56	90.5
12	9	1	3	35.0	SUMMER	33	24	35	56	90.5
15	8	1	4	35.0	SUMMER	33	24	35	56	90.5
16	6	1	5	35.0	SUMMER	33	24	35	56	90.5
17	3	1	6	35.0	SUMMER	33	24	35	56	90.5
13	2	1	7	35.0	SUMMER	33	24	35	56	90.5
9	6	1	8	35.0	SUMMER	33	24	35	56	90.5
2	4	1	1	56.0	WINTER	33	24	35	56	90.5
4	6	1	2	56.0	WINTER	33	24	35	56	90.5
9	9	1	3	56.0	WINTER	33	24	35	56	90.5
12	7	1	4	56.0	WINTER	33	24	35	56	90.5
14	5	1	5	56.0	WINTER	33	24	35	56	90.5
14	3	1	6	56.0	WINTER	33	24	35	56	90.5
9	1	1	7	56.0	WINTER	33	24	35	56	90.5
2	4	1	8	56.0	WINTER	33	24	35	56	90.5
1	2	1	1	90.5	ALL	33	24	35	56	90.5
2	5	1	2	90.5	ALL	33	24	35	56	90.5
9	9	1	3	90.5	ALL	33	24	35	56	90.5
12	9	1	4	90.5	ALL	33	24	35	56	90.5
12	9	1	5	90.5	ALL	33	24	35	56	90.5
15	8	1	6	90.5	ALL	33	24	35	56	90.5
16	6	1	7	90.5	ALL	33	24	35	56	90.5
16	6	1	8	90.5	ALL	33	24	35	56	90.5
17	3	1	9	90.5	ALL	33	24	35	56	90.5
13	2	1	10	90.5	ALL	33	24	35	56	90.5
9	1	1	11	90.5	ALL	33	24	35	56	90.5
5	1	1	12	90.5	ALL	33	24	35	56	90.5
1	2	1	13	90.5	ALL	33	24	35	56	90.5

FIGURE 1

HOME RANGE PLOT

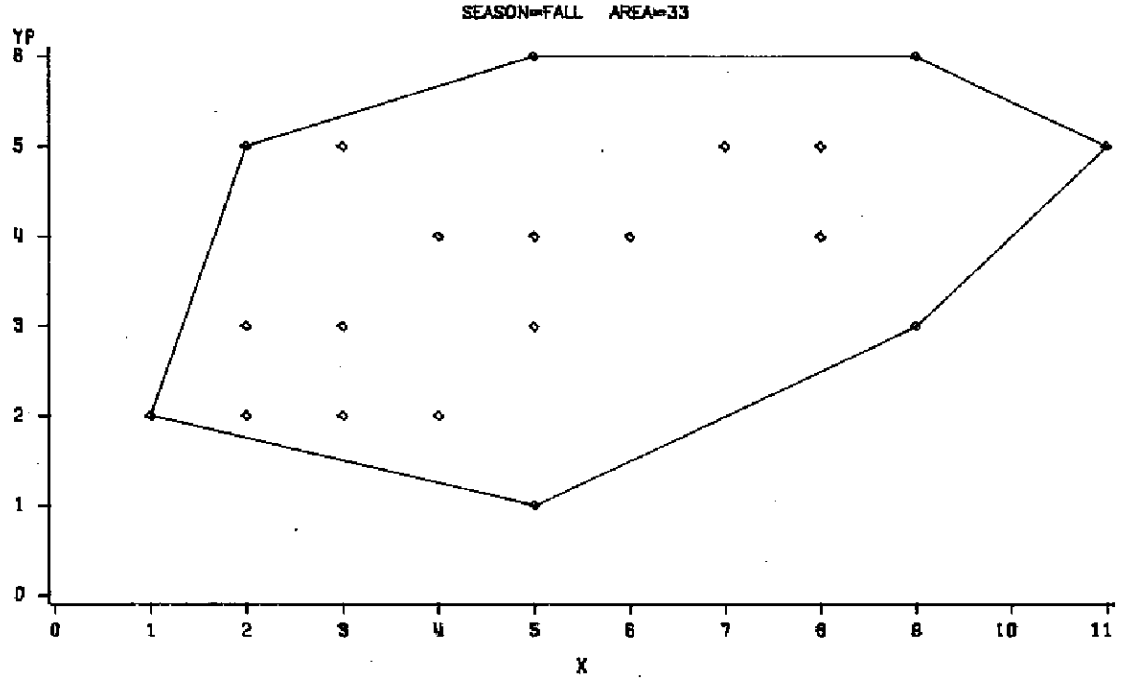
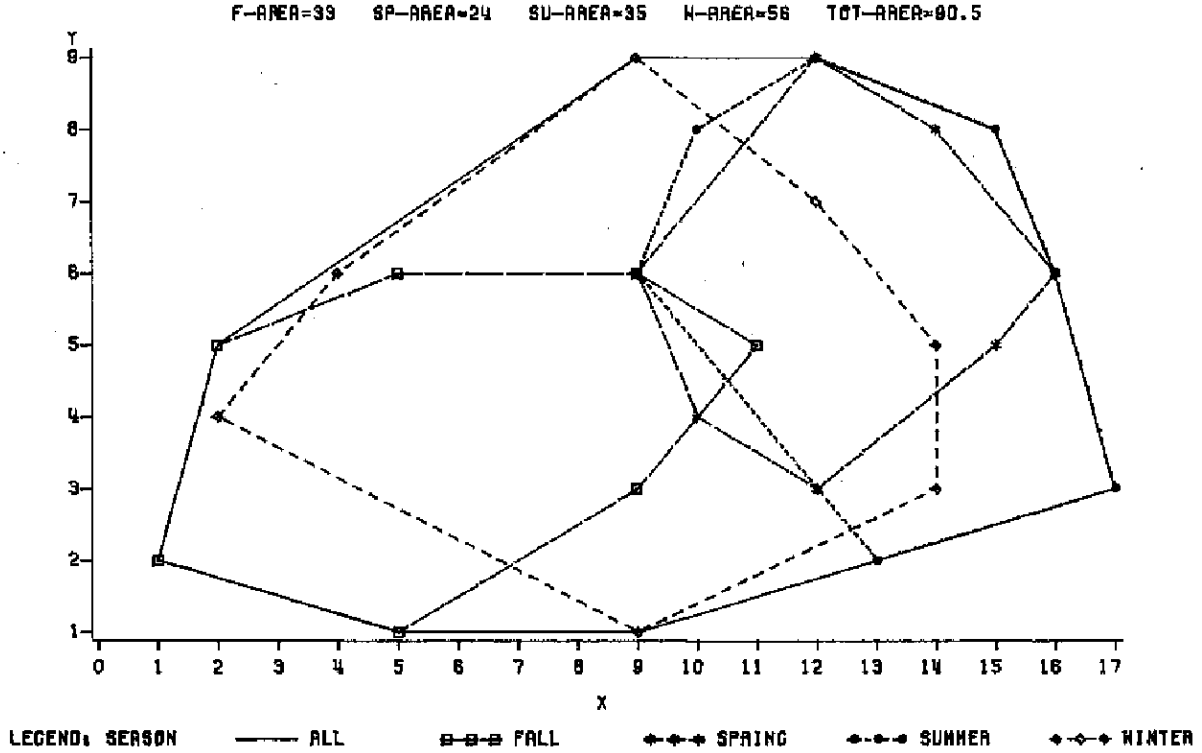


FIGURE 2

HOME RANGE PLOT



APPENDIX 1.

MACRO RANGE

```
PROC SORT; BY X Y; PROC MATRIX; FETCH XI DATA=IN;
*                                                    ;
*  INITILIZATION                                     ;
*                                                    ;
  J=1; N=NROW(XI); IV=J.(N,3,0);    X = XI || IV;
  FREE IV; FIRST=1; M=1;  AREA=0; INCR=1;
*                                                    ;
* DETERMINE THE GREATEST SLOPE                       ;
*                                                    ;
      DO WHILE(J NE 1 OR FIRST);
         FIRST=0;  I=J+INCR; L=0; TSLOPE=-999999;
*                                                    ;
         DO WHILE(1 LE I AND I LE N);
*                                                    ;
              IF((X(I,1)-X(J,1)) EQ 0)
                THEN SLOPE=9999999;
                ELSE  SLOPE= (X(I,2)-X(J,2)) #/
                             (X(I,1)-X(J,1)) ;
*                                                    ;
              IF(SLOPE GT TSLOPE)THEN DO;
                L=I; TSLOPE=SLOPE;
              END;
*                                                    ;
              I=I+INCR;
         END;
*                                                    ;
* MARK THE X,Y BOUNDARY PAIR, GIVE ENTRY ORDER       ;
*                                                    ;
         IF(L NE 0 AND X(L,3)=0 ) THEN DO;
           M=M+1; X(L,3)=1; X(L,4)=M;
*                                                    ;
* COMPUTE THE TRAPAZOIDAL AREA UNDER THE LAST 2      ;
*     POINTS AND ADD OR SUBTRACT IT                  ;
*          FROM THE TOTAL AREA                       ;
*                                                    ;
           TAREA=0;
           DELTAX=ABS( X(L,1)-X(J,1) );
           DELTAY=ABS( X(L,2)-X(J,2) );
           IF DELTAX NE 0 THEN DO;
              IF DELTAY EQ 0 THEN
                 TAREA=DELTAX*X(J,2);
              ELSE
                 TAREA=DELTAX*( X(J,2)><X(L,2) )
                       +.5*DELTAX*DELTAY;
           END;
           AREA=AREA+INCR*TAREA;
         END;
*                                                    ;
         J=L;      IF(J EQ N) THEN INCR=-1;
      END;
*                                                    ;
DO K=1 TO N;     X(K,5)=AREA;           END;

*                                                    ;
* SORT BY ORDER OF ENTRY - REPEATING FIRST           ;
*               BOUNDARY POINT                       ;
*                                                    ;
V=J.(1,5,0); X=X // V; X(N+1,)=X(1,);
X(N+1,4)=1;  XS=X;
XS(RANK(X(,4)),4)=X(,4); XS(RANK(X(,4)),3)=X(,3);
XS(RANK(X(,4)),2)=X(,2); XS(RANK(X(,4)),1)=X(,1);
X=XS; FREE XS;
*                                                    ;
OUTPUT X OUT=BOUND (RENAME=(COL1=X  COL2=Y
COL3=BOUND  COL4=ENTRY  COL5=AREA)) %
```

APPENDIX 2.

```
DATA ONE;      INPUT X Y SEASON $;       CARDS;
 1  2  FALL
                      2  5  FALL
PROC SORT; BY SEASON;
DATA IN; SET ONE; IF SEASON='FALL  ';
  RANGE;
   DATA FALL; SET BOUND; SEASON='FALL  ';
     DATA A1; SET FALL; IF _N_=1; I=1;
                              F_AREA=AREA;
              DROP X Y BOUND ENTRY AREA SEASON;
DATA IN; SET ONE; IF SEASON='SPRING';
  RANGE;
   DATA SPRING; SET BOUND; SEASON='SPRING';
                              P_AREA=AREA;
              DROP X Y BOUND ENTRY AREA SEASON;
DATA IN; SET ONE; IF SEASON='SUMMER';
  RANGE;
   DATA SUMMER; SET BOUND; SEASON='SUMMER';
     DATA A3; SET SUMMER; IF _N_=1; I=1;
                              SU_AREA=AREA;
              DROP X Y BOUND ENTRY AREA SEASON;
DATA IN; SET ONE; IF SEASON='WINTER';
  RANGE;
   DATA WINTER; SET BOUND; SEASON='WINTER';
     DATA A4; SET WINTER; IF _N_=1; I=1;
                              W_AREA=AREA;
              DROP X Y BOUND ENTRY AREA SEASON;
DATA IN; SET ONE; RANGE; DATA ALL; SET BOUND;
                              SEASON='ALL   ';
     DATA A5; SET ALL; IF _N_=1; I=1;
                              TOT_AREA=AREA;
              DROP X Y BOUND ENTRY AREA SEASON;
DATA COMBO; SET FALL SPRING SUMMER WINTER ALL;
IF BOUND=0 THEN DELETE; I=1;
DATA FINAL; MERGE COMBO(IN=Y1) A1(IN=Y2)
 A2(IN=Y3) A3(IN=Y4) A4(IN=Y5) A5(IN=Y6); BY I;
 IF Y1 & Y2 & Y3 & Y4 & Y5 & Y6; DROP ROW I;
PROC PRINT;
TITLE 'TABLE 2:    DATA SET FOR PLOTTING
          MULTIPLE HOME RANGES';
*********************************************;
GOPTIONS DEVICE=CAL936 HSIZE=10 VSIZE=7 GOPT10;
PROC GPLOT; BY F_AREA SP_AREA SU_AREA
       W_AREA TOT_AREA;
PLOT Y*X=SEASON;
SYMBOL1 I=JOIN V=NONE L=1;
SYMBOL2 I=JOIN V=SQUARE L=5;
SYMBOL3 I=JOIN V=STAR L=4;
SYMBOL4 I=JOIN V=+ L=2;
SYMBOL5 I=JOIN V=DIAMOND L=20;
TITLE1 .F=TRIPLEX .H=3 'HOME RANGE PLOT';
TITLE2;
```