Pure

Bangor University

DOCTOR OF PHILOSOPHY

An economic approach to assessing the value of recreation with special reference to forest areas.

Christensen, Jens Bjerregaard

Award date:
1985

Awarding institution:
Bangor University

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 19. Apr. 2024

85:S80

AN ECONOMIC APPROACH TO ASSESSING THE VALUE OF RECREATION WITH SPECIAL REFERENCE TO FOREST AREAS.

Jens Bjerregaard Christensen
(Cand. silv.)

APPENDICES:

A.4.1 Computer Programme Clawson

A.5.2 Computer Programme PFAFO1.FOR
A.5.3 Computer Programme PFAFO2.FOR and PFAFO3.FOR
A.5.4 Example of Output

I'W DDEFNYDDIO YN Y
LLYFRGELL YN UNIG

TO BE CONSULTED IN THE
LIBRARY ONLY

BEST COPY

AVAILABLE

Variable print quality

Text cut off in original

DAMAGED

TEXT

IN

ORIGINAL

.Appendix A4.1

.Computer Programme Clawson.

The programme consists of a main programme CLAW5 in file CLAW.FOR which calls subroutines from the five subfiles SUB15.FOR to SUB55.FOR

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

```
FFFFF   333    4   4    999                    J   HBBB     CCCC   H    H
F      3   3   4   4   9   9                   J   B   B   C       H    H
F          3   4   4   9   9                   J   B   B   C       H    H
FFFF       3   44444   9999                    J   BBBB    C       HHHHH
F          3       4       9               J   J   B   B   C       H    H
F      3   3       4       9               J   J   B   B   C       H    H
F       333        4    999                 JJJ    BBBB     CCCC   H    H
```

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

```
N    N   000   TTTTT  EEEEE                CCCC  L       AAA   W   W   SSSS   000
N    N  0   0    T    E      ::           C      L      A   A  W   W  S      0   0
NN   N  0   0    T    E      ::           C      L      A   A  W   W  S      0   0
N N  N  0   0    T    EEEE                C      L      A   A  W   W   SSS   0   0
N  N N  0   0    T    E      ::           C      L      AAAAA  W W W      S  0   0
N    N  0   0    T    E      ::           C      L      A   A  WW WW      S  0   0
N    N   000    T    EEEEE                CCCC  LLLLL  A   A  W   W  SSSS   000
```

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

START User F349 JBCH [146,24155] Job CLAW5 Seq. 1516 Date 13-Aug-81 12:12:12

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000111111111122222222233333333334444444444555555555566666666667777777778888888888999999999900000000001111111111222222
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456
```

```
00100           PROGRAM CLAW12
00200     C                        ********* CLAW5.FOR **********
00300     C
00400     C
00500     C                      RUN WITH SUB15,SUB25,SUB35,SUB45,SUB55
00600     C
00700     COMMENT
00800     C
00900     C     A PROGRAM FOR CLAWSON ANALYSIS  (VERSION 1.2)
01000     C
01100     COMMENT
01200           DIMENSION  XPZO(21),YPRI(21),TIT1(10),TIT2(10),IZOP(21),
01300          1 ITZO(21),TIT3(10),XAR2(21),YAR2(21)
01400           DOUBLE PRECISION NAME1,NAME2,NAME3
01500           LOGICAL BATCH,MAT,SUBR
01600           COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT/VAR1/XRES(20),
01700          1 XRES2(20),XC2F(3)
01800           COMMON/WEIA/WEI1(21),WEI2(21),WEI3(21),WEI4(21),WEI5(21),WEI6(21)
01900          1 ,WEI7(21)
02000           CALL ERRSET(0)
02100           MAT=.FALSE.
02200           SUBR=.FALSE.
02300           ITTX=5
02400           ITTY=5
02500           IDUM=2
02600           IPLN=1
02700           WRITE(ITTY,100)
02800     100   FORMAT(///,4X,'CLAWSON PROGRAMME ENTERED',//)
02900     110   CONTINUE
03000     C
03100     C     BATCH INFORMATION ; IF INTERACTIVE GOTO 700
03200     C
03300     190   WRITE(ITTY,200)
03400     200   FORMAT(/,4X,'PROGRAM TO BE RUN INTERACTIVE (YES OR NO) ? ',
03500          1  $)
03600           READ(ITTX,300,ERR=190)ANSWO
03700     300   FORMAT(A3)
03800           BATCH=.FALSE.
03900           IF (ANSWO.EQ.'YES') GO TO 700
04000           BATCH=.TRUE.
04100     310   WRITE(ITTY,400)
04200     400   FORMAT(/,4X,'INSTRUCTION FILE NAME ? ',$)
04300           READ(ITTX,500,ERR=310)NAME1
04400     500   FORMAT(A10)
04500     540   WRITE(ITTY,550)
04600     550   FORMAT(/,4X,'OUTPUT TO FILE (YES OR NO) ? ',$)
04700           READ(ITTX,300,ERR=540)ANSW2
04800           IF (ANSW2.NE.'YES') GO TO 650
04900     580   WRITE(ITTY,600)
05000     600   FORMAT(/,4X,'OUTPUT FILE NAME ? ',$)
05100           READ(ITTX,500,ERR=580)NAME2
05200     C
05300     C     OUTPUT ON UNIT 24 IF BATCH
05400     C
05500           OPEN(UNIT=24,FILE=NAME2)
05600           ITTY=24
05700     650   CONTINUE
05800     C
05900     C     INPUT ON UNIT 23 IF BATCH
06000     C
```

```
06100         OPEN(UNIT=23,FILE=NAME1,ACCESS='SEQIN')
06200         ITTX=23
06300   790   WRITE(5,800)
06400   800   FORMAT(/,4X,'PLOTS WANTED (YES OR NO) ? ',$)
06500         READ(5,800,ERR=790)ANSW1
06600         IF (ANSW1.EQ.'YES') IDUM=1
06700   700   CONTINUE
06800   C
06900   C     IF MATRIX AVAILABLE IT CAN BE READ IN HERE
07000   C
07100         IF (BATCH) GO TO 710
07200         WRITE(ITTY,702)
07300   702   FORMAT(/,4X,'MATRIX AVAILABLE (YES OR NO) ? ',$)
07400         READ(ITTX,300)ANSW7
07500         IF (ANSW7.EQ.'YES') MAT=.TRUE.
07600         IF (.NOT.MAT) GO TO 710
07700         CALL MATIN(XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
07800         GO TO 720
07900   C
08000   C  CALL INOUT ,WHICH PROVIDES THE MAIN PROGRAM WITH THE INITIAL
08100   C                 INFORMATION :POINTS FOR TRIP DEMAND CURVE
08200   C                             :ZONES,ZONEPOPULATION ETC.
08300   C
08400   710   CALL INOUT(XPZO,YPRI,IZON,IZOP,ITZO,ITOT,BATCH,NAME3)
08500   C
08600   C REDUCES THE POINTS IF THERE IS A ZONE WITHOUT ANY VISITORS
08700   C
08800   720   CALL REDUC(1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
08900   C
09000   C CREATES THE WEIGHTS FOR WEIGHTED ANALYSIS
09100   C
09200         CALL WEIGHT(IZON,XPZO,ITZO,IZOP)
09300         IF (.NOT.BATCH) ITTX=5
09400   C
09500   C   PLOTTING THE INITIAL POINTS FOR THE TRIP DEMAND CURVE
09600   C
09700         IF (BATCH) GO TO 2125
09800   6     WRITE(ITTY,2110)
09900   2110  FORMAT(/,4X,'PLOT OF POINTS FOR TRIP DEMAND CURVE (YES OR NO) ? ',
10000        .1$)
10100   7     READ(ITTX,300,ERR=6)ANSW4
10200         IF (ANSW4.EQ.'YES') IDUM=1
10300   2125  CONTINUE
10400         IF (IDUM.EQ.2) GO TO 2300
10500         WRITE(ITTY,2200)
10600   2200  FORMAT(/,4X,'PLOT OF TRIP DEMAND CURVE:',/,4X,'**********************
10700        1*********',/)
10800         DATA III1/'POINTS FOR TRIP DEMAND CURVE*.'/
10900         NA1='*.'
11000         DATA III3/'NO OF VISITORGROUPS/'/
11100   C
11200   COMMENT NON STANDARD FORTRAN
11300   C
11400         ENCODE(10,2250,TIT3(5))K1,NA1
11500   2250  FORMAT(I8,A2)
11600         CALL PLOT1(1,XPZO,YPRI,IZON,2,XPZO,YPRI,IZON,TIT1,6,5,TIT3,6,5,1)
11700         DATA III2/'LINE THROUGH POINTS*.'/
11800         IFAI2=1
11900         CALL SORT1(XPZO,YPRI,IZON,XAR2,YAR2,IFAI2,ITTY)
12000         CALL PLOT1(2,XPZO,YPRI,IZON,2,XAR2,YAR2,IZON,TIT1,6,5,TIT2,6,5,2)
```

```
12100           IF (.NOT.BATCH) IDUM=2
12200     2300 CONTINUE
12300     COMMENT
12400     C
12500     C   FITTING THE TRIP DEMAND CURVE AND CALCULATING AGGREGATED DEMAND
12600     C
12700     COMMENT
12800           IF (BATCH) GO TO 3100
12900     2900 CONTINUE
13000           IDUM=2
13100     C
13200     C  FOR INTERACTIVE VERSION PROMPTS FOR KIND OF ANALYSIS
13300     C
13400           WRITE(ITTY,3002)
13500     3002 FORMAT(/,4X,'TYPE OF ANALYSIS  ? ',$)
13600           GO TO 3003
13700     2990 SKIP RECORD ITTX
13800           WRITE(ITTY,3000)
13900     3000 FORMAT(/,4X,'WHAT KIND OF ANALYSIS (1 TO 3):',/
14000          1          ,6X,'1  : LINEAR REGRESSION',/
14100          2          ,6X,' 12: LINEAR REG. (WEIGHTED TO ZONEPOP) ',/
14200          2          ,6X,' 13: LINEAR REG. (WEIGHTED TO ZONEPOP/VISITRATE)  ',/
14300          2          ,6X,' 14: LINEAR REG. (SUBJECTIVE WEIGHTED) ',/
14400          2          ,6X,' 15: LINEAR REG. (WEIGHTED TO NO. OF VISITORGROUPS)',
14500          2   /      ,6X,' 16: LINEAR REG. (WEIGHTED SQRT(ZPOP)/(NO OF VGROUPS)
14600          2)',/      ,6X,' 17: LINEAR REG. (WEIGHTED TO SQRT(ZONEPOP))  ',/
14700          3          ,6X,'2  : 2ND DG POLYNOMIAL ',/
14800          4          ,6X,' 22: 2ND DG POL. (WEIGHTED TO ZONEPOP)',/
14900          4          ,6X,' 23: 2ND DG POL. (WEIGHTED TO ZONEPOP/VISITRATE)',/
15000          4          ,6X,' 24: 2ND DG POL. (SUBJECTIVE WEIGHTED) ',/
15100          2          ,6X,' 25: 2ND DG POL. (WEIGHTED TO NO. OF VISITORGROUPS)',
15200          2   /      ,6X,' 26: 2ND DG POL. (WEIGHTED SQRT(ZPOP)/(NO OF VGROUPS)
15300          2)',/      ,6X,' 27: 2ND DG POL. (WEIGHTED TO SQRT(ZONEPOP)) ',/
15400          3          ,6X,'3  : TRANSFORMED LIN. REG. ',/
15500          4          ,6X,' 32: TRNF. LIN. REG. (WEIGHTED TO ZONEPOP)',/
15600          4          ,6X,' 33: TRNF. LIN. REG. (WEIGHTED TO ZONEPOP/VISITRATE)'
15700          4,/        ,6X,' 34: TRNF. LIN. REG. (SUBJECTIVE WEIGHTED) ',/
15800          2          ,6X,' 35: TRNF. LIN. REG. (WEIGHTED TO NO. OF VISITORGROUP
15900          2S)',/
16000          '2         ,6X,' 36: TRNF. LIN. REG. (WEIGHTED SQRT(ZPOP)/(NO OF VGRO
16100          2UPS)',/
16200          2          ,6X,' 37: TRNF. LIN. REG. (WEIGHTED TO SQRT(ZONEPOP)) ',/
16300          8          ,6X,'4  : NO MORE ANALYSES ',/
16400          9          ,32X,' ? ',$)
16500     3003 READ(ITTX,3004,ERR=2990)INAN
16600     3004 FORMAT(I5)
16700           IF ((INAN.LT.1).OR.(INAN.GT.39)) GO TO 3018
16800           IF (INAN.EQ.4) GO TO 3012
16900     3005  WRITE(ITTY,3010)
17000     3010 FORMAT(/,4X,'PLOTS WANTED (YES OR NO) ? ',$)
17100           READ(ITTX,300,ERR=3005)ANSW3
17200           IF (ANSW3.EQ.'YES') IDUM=1
17300     3012 GO TO (3100,3300,3500,4000,3019,3019,3019,3019,3019,3100,
17400          2      3100,3250,3230,3240,3200,3220,3210,3019,3019,3300,
17500          3      3300,3450,3430,3440,3400,3420,3410,3019,3019,3019,
17600          4      3500,3650,3630,3640,3600,3620,3610,3019,3019,3019)INAN
17700     3018 CONTINUE
17800     3019 WRITE(ITTY,3020)
17900     3020 FORMAT(/,4X,'WRONG NUMBER OF ANALYSIS !!!')
18000          GOTO 2900
```

```
C
C       LINEAR REGRESSION
C
 5100 CALL GLM2(1,YPRI,XPZO,IZON,WEI1)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C       WEIGHTED LINEAR REGRESSION (ACCORDING TO NO OF VISITORGROUPS)
C
 5200 CALL GLM2(2,YPRI,XPZO,IZON,WEI4)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C       WEIGHTED LINEAR REGRESSION (ACCORDING TO SQRT(ZONEPOPULATION) )
C
 5210 CALL GLM2(3,YPRI,XPZO,IZON,WEI2)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C WEIGHTED LINEAR REGRESSION (SQRT(ZONEPOP)*NO OF VISITORGROUPS
C
 5220 CALL GLM2(4,YPRI,XPZO,IZON,WEI5)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C       WEIGHTED LINEAR REGRESSION  (ZONEPOP/VISITRATE)
C
 5230 CALL GLM2(5,YPRI,XPZO,IZON,WEI3)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
      IF (BATCH) GO TO 3290
C
C       WEIGHTED LINEAR REGRESSION (SUBJECTIVE)
C
 3240 CALL SUBWEI(IZON,WEI7,SUBW)
      CALL GLM2(6,YPRI,XPZO,IZON,WEI7)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
 3290 CONTINUE
C
C       WEIGHTED LINEAR REGRESSION (ACCORDING TO ZONEPOPULATION )
C
```

```
 3250 CALL GLR2(7,YPRI,XPZO,IZON,WEI6)
      CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C     2ND DG POLYNOMIAL
C
 3300 CALL LSFI2(1,YPRI,XPZO,IZON,WEI1)
      CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C     WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO NO OF VISITORS)
C
 3400 CALL LSFI2(2,YPRI,XPZO,IZON,WEI4)
      CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C     WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO SQRT(ZONEPOPULATION) )
C
 3410 CALL LSFI2(3,YPRI,XPZO,IZON,WEI2)
      CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C     WEIGHTED 2ND DG POLYNOMIAL ( SQRT(ZONEPOP)*NO OF VISITORGROUPS )
C
 3420 CALL LSFI2(4,YPRI,XPZO,IZON,WEI5)
      CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C       WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO ZONEPOP/VISITRATE)
C
 3430 CALL LSFI2(5,YPRI,XPZO,IZON,WEI3)
      CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
      IF (BATCH) GO TO 3490
C
C       WEIGHTED 2ND DG POLYNOMIAL (SUBJECTIVE)
C
 3440 CALL SUBWEI(IZON,WEI7,SUBW)
      CALL LSFI2(6,YPRI,XPZO,IZON,WEI7)
      CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
```

```
30100          CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
30200          IF (.NOT.BATCH) GO TO 2900
30300    3490 CONTINUE
30400    C
30500    C      WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO ZONEPOPULATION )
30600    C
30700    3450 CALL LSFIT(7,YPRI,XPZO,IZON,WEI6)
30800          CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
30900          IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
31000         1 ,ITZO,ITOT)
31100          CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
31200          IF (.NOT.BATCH) GO TO 2900
31300    C
31400    C      IRNF LIN. REG.
31500    C
31600    3500 CALL IRNF(1,YPRI,XPZO,IZON,WEI1)
31700          CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
31800          IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
31900         1 ,ITZO,ITOT)
32000          CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
32100          IF (.NOT.BATCH) GO TO 2900
32200    C
32300    C      WEIGHTED IRNF LIN. REG. (ACCORDING TO NO OF VISITORGROUPS)
32400    C
32500    3600 CALL IRNF(2,YPRI,XPZO,IZON,WEI4)
32600          CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
32700          IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
32800         1 ,ITZO,ITOT)
32900          CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
33000          IF (.NOT.BATCH) GO TO 2900
33100    C
33200    C      WEIGHTED IRNF LIN. REG. (ACCORDING TO SQRT(ZONEPOPULATION) )
33300    C
33400    3610 CALL IRNF(3,YPRI,XPZO,IZON,WEI2)
33500          CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
33600          IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
33700         1 ,ITZO,ITOT)
33800          CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
33900          IF (.NOT.BATCH) GO TO 2900
34000    C
34100    C WEIGHTED IRNF LIN. REG. (SQRT(ZONEPOP)*NO OF VISITORGROUPS
34200    C
34300    3620 CALL IRNF(4,YPRI,XPZO,IZON,WEI5)
34400          CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
34500          IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
34600         1 ,ITZO,ITOT)
34700          CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
34800          IF (.NOT.BATCH) GO TO 2900
34900    C
35000    C      WEIGHTED IRNF LIN. REG.  (ZONEPOP/VISITRATE)
35100    C
35200    3630 CALL IRNF(5,YPRI,XPZO,IZON,WEI3)
35300          CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
35400          IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
35500         1 ,ITZO,ITOT)
35600          CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
35700          IF (.NOT.BATCH) GO TO 2900
35800          IF (BATCH) GO TO 3690
35900    C
36000    C      WEIGHTED TRNF LIN. REG. (SUBJECTIVE)
```

```
C
 3640 CALL SUBREI(IZON,REI7,SUBN)
      CALL TRNF(6,YPRI,XPZO,IZON,REI7)
      CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
 3690 CONTINUE
C
C     WEIGHTED TRNF LIN. REG. (ACCORDING TO ZONEPOPULATION )
C
 3650 CALL TRNF(7,YPRI,XPZO,IZON,REI6)
      CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
     1 ,ITZO,ITOT)
      CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
      IF (.NOT.BATCH) GO TO 2900
C
C     TERMINATING THE PROGRAMME
C
      IF (BATCH) GO TO 4030
 4000 CONTINUE
      IF (.NOT.BATCH) IDUM=2
      GO TO 4006
 4005 SKIP RELOAD ITTX
 4008 WRITE(ITTY,4010)
 4010 FORMAT(/,4X,'OPTIONS : ',/
     1       ,6X,'1: BEGIN WITH NEW DATA ',/
     2       ,6X,'2: EXIT FROM PROGRAMME',/
     3       ,30X,'? ',$)
      READ(ITTX,*,ERR=4005)IANS
      GO TO (110,4030)IANS
      GO TO 2900
 4030 CONTINUE
C
C     CLOSE AND DELETE TMP-FILE
C
      WRITE(ITTY,2800)
 2800 FORMAT(///,4X,'EXIT FROM CLARSON PROGRAMME',//,4X,'C*jbch1981',//)
      CLOSE(UNIT=22,FILE=NAME3,DISPOSE='DELETE')
      IF (BATCH) CLOSE(UNIT=23,FILE=NAME1,DISPOSE='SAVE')
      IF (BATCH) CLOSE(UNIT=24,FILE=NAME2,DISPOSE='SAVE')
      STOP
      END
```

```
00100    C                        ******* SUB15.FOR *********
00200    C
00300    C                        SUBROUTINE INOUT  MATIN
00400    C
00500    C
00600    C
00700          SUBROUTINE INOUT(XPZO,YPP1,IZON,IZOP,ITZO,ITOT,BATCH,NAME3)
00800    COMMENT
00900    C
01000    C   SUBROUTINE   INOUT
01100    C                                   XPZO: NO/K1 VISITING
01200    C                                   YPRI: PRICE FROM ZONE (AVERAGE OR FIXED)
01300    C                                   IZON: NO OF ZONES
01400    C                                   IZOP: POPULATION IN ZONE
01500    C                                   ITZO: NO OF VISITORGROUPS FROM ZONE
01600    C                                   BATCH  : LOGICAL
01700    C
01800    C                                   SEE DIMENSION STATEMENT FOR ARRAYS
01900    COMMENT
02000          DIMENSION NAM(10),X(10),IZOP(21),ITZO(21),IZOO(21),
02100         1  XPZO(21),YPPI(21),YPR2(21),IDEN(21),XMEN(21),YPR3(21)
02200          COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT
02300          DOUBLE PRECISION NAME1,XPZ01(21),XMAX,XMIN,NAME3,NAME2
02400          LOGICAL BATCH,TRAD,NSUC,EZON,RWEI
02500     140  CONTINUE
02600          TRAD=.FALSE.
02700          EZON=.FALSE.
02800          RWEI=.FALSE.
02900    COMMENT
03000    C
03100    C  INFORMATION ON RAW DATA FILE
03200    C     NAME OF FILE,NO OF VARIABLES,NAME OF VARIABLES,NO OF OBSERVATIONS
03300    C
03400    COMMENT
03500          GO TO 159
03600     158  SKIP RECORD ITTX
03700     159  WRITE(ITTY,160)
03800     160  FORMAT(///,4X,'TYPE OF ANALYSIS (1 TO 3) : ',/
03900         1          6X,' 1: TRADITIONAL CLARKSON',/
04000         2          6X,' 2: REVISED CLARKSON',  /
04100         3          6X,' 3: REV. CLARKSON (WEIGH.)',/
04200         4          6X,'                          ? ',$)
04300          READ(ITTX,*,ERR=158)IANA
04400     166  FORMAT(4X,I2,/)
04500          IF (BATCH) WRITE(ITTY,166)IANA
04600          IF (IANA.EQ.1) TRAD=.TRUE.
04700          IF (IANA.EQ.3) RWEI=.TRUE.
04800          GO TO 169
04900     168  SKIP RECORD ITTX
05000     169  WRITE(ITTY,170)
05100     170  FORMAT(//,4X,'ZONING BASED ON (1 OR 2) :',/
05200         1          ,6X,' 1: DISTANCE  ',/
05300         2          ,6X,' 2: ZONE IDENTIFIER ',/
05400         3          ,30X,'? ',$)
05500          READ(ITTX,*,ERR=168)IAN2
05600          IF (BATCH) WRITE(ITTY,166)IAN2
05700          IF (IAN2.EQ.2) EZON=.TRUE.
05800          GO TO 191
05900     190  SKIP RECORD ITTX
06000     191  WRITE(ITTY,200)
```

```
06100     200  FORMAT(/,4X,'RAW DATA FILE ? ',$)
06200          READ(ITIX,220,ERR=190)NAME1
06300     220  FORMAT(A10)
06400     300  FORMAT(4X,A10,/)
06500          IF (BATCH) WRITE(ITTY,300)NAME1
06600     C
06700     C  UNIT 20 : RAW DATA FILE
06800     C
06900          OPEN(UNIT=20,FILE=NAME1,ACCESS='SEQIN')
07000          GO TO 390
07100     350  SKIP RECORD ITTX
07200     390  WRITE(ITTY,400)
07300     400  FORMAT(4X,'HOW MANY VARIABLES TO BE READ (MAX 10) ? ',$)
07400          READ(ITIX,*,ERR=350)NUM
07500     450  FORMAT(4X,I2,/)
07600          IF (BATCH) WRITE(ITTY,450)NUM
07700          DO 455 I=1,NUM
07800          WRITE(ITTY,451)I
07900     451  FORMAT(4X,'NAME OF VARIABLE NUMBER',X,I2,X,'? ',$)
08000          READ(ITIX,454)NAM(I)
08100     454  FORMAT(A5)
08200     452  FORMAT(4X,A5,/)
08300          IF (BATCH) WRITE(ITTY,452)NAM(I)
08400     455  CONTINUE
08500          GO TO 472
08600     470  SKIP RECORD ITTX
08700     472  WRITE(ITTY,475)
08800     475  FORMAT(4X,'HOW MANY OBSERVATIONS ? ',$)
08900          READ(ITIX,*,ERR=470)NUMO
09000      480  FORMAT(4X,I6,/)
09100           IF (BATCH) WRITE(ITTY,480)NUMO
09200     COMMENT
09300     C
09400     C  SUMMARY OF INPUT
09500     C
09600     COMMENT
09700      3000 CONTINUE
09800           WRITE(ITTY,4000)NAME1
09900      4000 FORMAT(5X,'SUMMARY OF INPUT:',/,5X,'*****************',/,/,7X,'RAW
10000         ,2 DATA FILE',6X,A10,/,/,7X,'NAMES OF VARIABLES :')
10100           DO 4004 I=1,NUM
10200           WRITE(ITTY,4003)NAM(I)
10300      4003  FORMAT(24X,A10)
10400      4004  CONTINUE
10500           WRITE(ITTY,4010)
10600      4010 FORMAT(/,4X,'ANY CHANGES (YES OR NO) ? ',$)
10700           READ(ITIX,225)ANSW2
10800      225  FORMAT(A3)
10900           IF (BATCH) WRITE(ITTY,4020)ANSW2
11000      4020 FORMAT(4X,A3,/)
11100           IF (ANSW2.EQ.'YES') GO TO 140
11200     COMMENT
11300     C
11400     C  INFORMATION ON ZONES
11500     C      NO OF ZONES,POPULATION IN ZONES
11600     C
11700     COMMENT
11800      496  CONTINUE
11900     C
12000     C    INITIALIZES THE VARIABLES
```

```
12100     C
12200           REMIND 20
12300           DO 150 I=1,21
12400           ITZJ(I)=0.0
12500           YPR1(I)=0.0
12600           YPR3(I)=0.0
12700           XOF1(I)=0.0
12800           IUEN(I)=0
12900           ITZJ(I)=0
13000           IZO3(I)=0
13100           IZOP(I)=0
13200     150   CONTINUE
13300           ITOT=0
13400           GO TO 490
13500     488   SKIP RECORD ITTX
13600     490   WRITE(ITTY,500)
13700     500   FORMAT(4X,'HOW MANY ZONES (MAX 20) ? ',$)
13800           READ(ITTX,*,ERR=488)IZON
13900     600   FORMAT(4X,I6,/)
14000           IF (BATCH) WRITE(ITTY,600)IZON
14100           IF (IZON.LE.0) GOTO 605
14200           DO 605 I=1,IZON
14300           IF (EZON) GO TO 650
14400           IF (I.EQ.IZON) GO TO 609
14500     602   CONTINUE
14600           GO TO 592
14700     591   SKIP RECORD ITTX
14800     592   WRITE(ITTY,601)I
14900     601   FORMAT(4X,'ZONE NO ',I2,' EXTENDS TO MILES ? ',$)
15000           READ(ITTX,*,ERR=591)IZOB(I)
15100           IF (BATCH) WRITE(ITTY,604)IZOB(I)
15200     606   CONTINUE
15300           GO TO 594
15400     593   SKIP RECORD ITTX
15500     594   WRITE(ITTY,603)I
15600     603   FORMAT(4X,'POPULATION IN ZONE NO ',I2,' ? ',$)
15700           READ(ITTY,*,ERR=593)IZOP(I)
15800     604   FORMAT(4X,I10,/)
15900           IF (BATCH) WRITE(ITTY,604)IZOP(I)
16000           IF (.NOT.TRAD) GO TO 605
16100           GO TO 596
16200     595   SKIP RECORD ITTX
16300     596   WRITE(ITTY,616)I
16400     616   FORMAT(4X,'TRAVEL COST FROM ZONE NO ',I2,' (PENCE) ? ',$)
16500           READ(ITTX,*,ERR=595)YPR2(I)
16600           IF (BATCH) WRITE(ITTY,618)YPR2(I)
16700     618   FORMAT(8X,F6.0,/)
16800     605   CONTINUE
16900           GO TO 650
17000     609   WRITE(ITTY,610)
17100     610   FORMAT(4X,'DOES THE OUTER ZONE HAVE ANY BOUNDARY (YES OR NO) ? ',$
17200          1)
17300           READ(ITTX,225)ANSWO
17400     615   FORMAT(4X,A3,/)
17500           IF (BATCH) WRITE(ITTY,615)ANSWO
17600           IF (ANSWO.EQ.'YES') GO TO 602
17700           IZOB(IZON)=10000
17800           GO TO 606
17900     650   CONTINUE
18000           GO TO 652
```

```
      631   SKIP RECORD ITTX
      632   WRITE(ITTY,652)I
      652   FORMAT(4X,'ZONE NO ',I2,' HAS IDENTIFICATION NO ? ',$)
            READ(ITTX,*,ERR=631)IDEN(I)
            IF (BATCH) WRITE(ITTY,653)IDEN(I)
      653   FORMAT(4X,I5,/)
            GO TO 606
      630   CONTINUE
C
C     SUMMARY OF INPUT ON ZONES
C
            WRITE(ITTY,640)
      640   FORMAT(1H1,/,5X,'SUMMARY OF INPUT:',/,5X,'******************',/)
            DO 4005 I=1,IZON
            IF (TRAD) GO TO 4007
            IF (EZON) GO TO 646
            WRITE(ITTY,4002)I,IZOB(I),IZOP(I)
      4002  FORMAT(5X,'ZONE NO ',I2,' UNTIL ',I5,' MILES ; POPULATION ',I8)
            GO TO 4008
       646  CONTINUE
            WRITE(ITTY,648)I,IDEN(I),IZOP(I)
       648  FORMAT(5X,'ZONE NO ',I2,'; IDENTIFICATION NO ',I5,'; POPULATION ',
           1  I8)
            GO TO 4008
      4007  CONTINUE
            IF (EZON) GO TO 660
            WRITE(ITTY,4006)I,IZOB(I),IZOP(I),YPR2(I)
      4006  FORMAT(5X,'ZONE NO ',I2,' UNTIL ',I5,' MILES ; POPULATION ',I8,
           1 '; TRAVEL COSTS',F8.0)
            GO TO 4008
       660  CONTINUE
            WRITE(ITTY,662)I,IDEN(I),IZOP(I),YPR2(I)
       662  FORMAT(5X,'ZONE NO ',I2,';IDENT.NO ',I5,';POPUL. ',I8,
           1     ';TR.COST ',F8.0)
      4008  CONTINUE
            WRITE(ITTY,670)
       670  FORMAT(/,4X,'ANY CHANGES IN THE ZONING (YES OR NO) ? ',$)
            READ(ITTX,225)ANS3
       671  FORMAT(4X,A3,/)
            IF (BATCH) WRITE(ITTY,671)ANS3
            IF (ANS3.EQ.'YES') GO TO 498
      COMMENT
C
C  CALCULATING X FROM EACH ZONE (XPZO,YPRI)
C
      COMMENT
            IF (.NOT.EZON) GO TO 4090
C
C       IDENTIFICATION OF VAR ON RAW DATA FILE
C
            GO TO 4079
      4078  SKIP RECORD ITTX
      4079  WRITE(ITTY,4080)
      4080  FORMAT(4X,'ZONE IDENTIFIER IS VARIABLE NO ? ',$)
            READ(ITTX,*,ERR=4078)KST
            IF (BATCH) WRITE(ITTY,4052)KST
            GO TO 4092
      4090  CONTINUE
            GO TO 4091
      4089  SKIP RECORD ITTX
```

```
      4091 WRITE(ITTY,4050)
      4050 FORMAT(4X,'MINEDISTANCE IS VARIABLE NO ? ',$)
           READ(ITTX,*,ERR=4089)IST
           IF (BATCH) WRITE(ITTY,4052)IST
      4092 CONTINUE
           IF (TRAD) GO TO 718
           GO TO 4059
      4058 SKIP RECORD ITTX
      4059 WRITE(ITTY,4060)
      4060 FORMAT(4X,'TRAVELCOSTS IS VARIABLE NO. ? ',$)
           READ(ITTY,*,ERR=4058)JST
      4052 FORMAT(4X,I2,/)
           IF (BATCH) WRITE(ITTY,4052)JST
      4037 GO TO 4039
      4038 SKIP RECORD ITTX
      4039 IF (.NOT.RREI) GO TO 718
           WRITE(ITTY,4040)
      4040 FORMAT(4X,'STAYLENGTH IS VARIABLE NO. ? ',$)
           READ(ITTX,*,ERR=4038)INE
           IF (BATCH) WRITE(ITTY,4052)INE
       718 CONTINUE
           ANSWS='NO'
           IF (((ANSWO.NE.'YES').OR.TRAD).OR.EZON)   GO TO 730
C
C          OBSERVATIONS OUTSIDE ZONES
C
           WRITE(ITTY,720)
       720 FORMAT(/,4X,'ARE OBSERVATIONS OUTSIDE THE OUTER ZONE TO BE INCLUDE
          1D (YES OR NO) ? ',$)
           READ(ITTX,225)ANSW5
           IF (BATCH) WRITE(ITTY,671)ANSW5
       733  CONTINUE
C
C     OPEN UNIT 22 AS WORKFILE FOR INDIVIDUAL INTEGRATION
C
           CALL RANNAM(1,NAME3)
           OPEN(UNIT=22,FILE=NAME3)
           DO 4200 I=1,NUMO
           READ(20,1000,ERR=733)(X(K),K=1,10)
      1000 FORMAT(10F)
           GO TO 734
       735 WRITE(ITTY,732)
       732 FORMAT(/,4X,'INPUT ERROR IN RAW DATA FILE  ! ',/
          1  ,20X,' START AGAIN WITH INPUT ',/)
           GO TO 140
       734 CONTINUE
           IF (.NOT.EZON) GO TO 4140
           NSUC=.TRUE.
           DO 4120 J=1,IZON
           K5=J
           IF (.NOT.(X(KST).EQ.IDEN(J))) GO TO 4120
           NSUC=.FALSE.
           GO TO 4122
      4120 CONTINUE
      4122 CONTINUE
           IF (NSUC) K5=K5+1
           ITZO(K5)=ITZO(K5)+1
           IF (.NOT.TRAD) YPRI(K5)=YPRI(K5)+X(JST)
           IF ((.NOT.TRAD).AND.(RREI)) XNEW(K5)=XNEW(K5)+(1/X(INE))
           IF ((.NOT.TRAD).AND.(RREI)) YPR3(K5)=YPR3(K5)+(X(JST)/X(INE))
```

```
      GO TO 4190
 4140 CONTINUE
      DO 4150 J=IZON,1,-1
      KS=J
      IF (X(IST).GT.IZOR(J)) GO TO 4170
 4150 CONTINUE
      ITZO(KS)=ITZO(KS)+1
      IF (.NOT.IRAD) YPRI(KS)=YPRI(KS)+X(JST)
      IF ((.NOT.IRAD).AND.(RNEI)) XNEW(KS)=XNEW(KS)+(1/X(IWE))
      IF ((.NOT.IRAD).AND.(RNEI)) YPR3(KS)=YPR3(KS)+(X(JST)/X(IWE))
      GO TO 4190
 4170 ITZO(KS+1)=ITZO(KS+1)+1
      IF (.NOT.IRAD) YPRI(KS+1)=YPRI(KS+1)+X(JST)
      IF ((.NOT.IRAD).AND.(RNEI)) XNEW(KS+1)=XNEW(KS+1)+(1/X(IWE))
      IF ((.NOT.IRAD).AND.(RNEI)) YPR3(KS+1)=YPR3(KS+1)+(X(JST)/X(IWE))
 4190 CONTINUE
      IF ((.NOT.IRAD).AND.((KS.LT.IZON).OR.((KS.EQ.IZON).AND.((ANSW5.EQ.
     1   'YES').OR.(EZON))))) WRITE(22,4194)X(JST)
 4194 FORMAT(F10.2)
 4200 CONTINUE                                                          A
      IZOP(1)=IZOR+1                                                    C
      DO 6000 I=1,IZON
      XPZO1(I)=SNGL(FLOAT(ITZO(I))/FLOAT(IZOP(I))
 6000 CONTINUE
      XMIN=XPZO1(1)
      XMAX=XPZO1(1)
      DO 6010 I=1,IZON
      XMIN=DMIN1(XMIN,XPZO1(I))
      XMAX=DMAX1(XMAX,XPZO1(I))
C
C       FINDS THE K1 VALUE ( NO/K1 INHABITANTS )
C
 6010 CONTINUE
      K1=INT(10.0/XMAX)
      K3=1
 6020 CONTINUE
      K2=(K1/(10**K3))
      IF (K2.EQ.0) GO TO 6030
      K3=K3+1
      GO TO 6020
 6030 CONTINUE
      K1=(10**K3)
C
C       OUTPUT RESULTS  ( XPZO YPRI )
C
      WRITE(ITTY,4053)
      WRITE(ITTY,4054)K1
 4053 FORMAT(1H1,/,5X,'RESULTS:',/,5X,'********')
 4054 FORMAT(/,10X,'NO. OF VISITORGROUPS /',I10,' INHABITANTS IN ZONE :'
     1 ,/)
      DO 4360 I=1,IZON
      XPZO(I)=SNGL(XPZO1(I)*K1)
      IF (IRAD) GO TO 6055
      IF (ITZO(I).NE.0.0) YPRI(I)=YPRI(I)/FLOAT(ITZO(I))
      IF ((RNEI).AND.(ITZO(I).NE.0.0)) YPRI(I)=YPR3(I)/XNEW(I)
 6055 CONTINUE
      IF (.NOT.IRAD) GO TO 4260
      YPRI(I)=YPR2(I)
      WRITE(ITTY,4252)I,XPZO(I),YPRI(I)
 4252 FORMAT(10X,'ZONE NO. ',I2,4X,F8.4,'    ; TRAVEL COSTS ',F8.0)
```

```
      GO TO 4300
4260  CONTINUE
      IF (PAEI) GO TO 4270
      WRITE(ITTY,4250)I,XPZO(I),YPRI(I)
4250  FORMAT(10X,'ZONE NO ',I2,4X,F8.4,'      ; AVERAGE COST ',F8.2)
      GO TO 4300
4270  CONTINUE
      WRITE(ITTY,4272)I,XPZO(I),YPRI(I)
4272  FORMAT(10X,'ZONE NO. ',I2,4X,F8.4,'   ; AVERAGE COST (WEIGH) ',
     1F8.2)
4300  CONTINUE
      XPCT=FLOAT(ITZO(IZON1))/FLOAT(NUMO)*100
C
C     OUTPUTS RESULTS OF OBSERVATIONS OUTSIDE ZONES
C
      IF (ANSW5.NE.'YES') GO TO 740
      IF (ITZO(IZON1).EQ.0) GO TO 4380
      WRITE(ITTY,4360)ITZO(IZON1),XPCT
4360  FORMAT(/,6X,'(',I4,' OBSERVATIONS, OR ',F5.2,' PCT. OF ALL OBSERVA
     1TIONS ARE OUTSIDE',/,10X,' THE OUTER ZONE, AND ARE GOING TO BE INC
     2LUDED AS POINT:)',/)
      XPZO(IZON1)=0.0
      YPRI(IZON1)=YPRI(IZON1)/FLOAT(ITZO(IZON1))
      IF (RAEI) YPRI(IZON1)=YPR3(IZON1)/XNEW(IZON1)
      WRITE(ITTY,4250)IZON1,XPZO(IZON1),YPRI(IZON1)
4380  CONTINUE
      WRITE(ITTY,4385)
4385  FORMAT(X)
      IZON=IZON1
      GO TO 844
740   CONTINUE
      IF (ITZO(IZON1).EQ.0) GO TO 844
      IF (EZON) GO TO 840
      WRITE(ITTY,4362)ITZO(IZON1),XPCT
4362  FORMAT(/,6X,'(',I4,' OBSERVATIONS, OR ',F5.2,' PCT. OF ALL OBSERVA
     1TIONS ARE OUTSIDE',/,10X,' THE OUTER ZONE, AND ARE EXCLUDED )')
      GO TO 844
840   CONTINUE
      WRITE(ITTY,842)ITZO(IZON1),XPCT
842   FORMAT(/,6X,'(',I4,' OBSERVATIONS, OR ',F5.2,
     1    ' PCT. OF ALL  ORSERVATIONS HAD AN UNDEFINED',/,10X,
     2    'ZONEIDENTIFIER, AND ARE EXCLUDED ! )')
844   CONTINUE
      WRITE(ITTY,845)
845   FORMAT(X)
      DO 4400 I=1,IZON
      WRITE(ITTY,4390)I,ITZO(I)
4390  FORMAT(10X,'NUMBER FROM ZONE ',I2,4X,';',I6,' VISITORGROUPS')
      ITOT=ITOT+ITZO(I)
4400  CONTINUE
      WRITE(ITTY,680)
 680  FORMAT(/,4X,'ANY CHANGES IN THE ZONING BEFORE FITTING THE CURVE (Y
     1ES OR NO) ? ',$)
      READ(ITTX,225)ANSW4
      IF (BATCH) WRITE(ITTY,671)ANSW4
      IF (ANSW4.EQ.'YES') GO TO 498
      CLOSE(UNIT=20,FILE=NAME1,DISPOSE='SAVE')
      WRITE(ITTY,6300)
6300  FORMAT(/,4X,'INFORMATION TO BE STORED IN MATRIX (YES OR NO) ? ',$
     1)
```

```
C2100          READ(ITTX,225)ANS..6
C2200          IF (BATCH) WRITE(ITTY,671)ANSW6
C2300          IF (ANS..6.NE.'YES') GO TO 6666
C2400    C
C2500    C       CREATING INPUT MATRICE IN FILE C????.MAT (UNIT=21)
C2600    C
C2700          CALL RAN.AM(2,NAME2)
C2800          OPEN(U.IT=21,FILE=NAME2)
C2900          WRITE(21,*)IZON,ITOT,K1
C3000          DO 6400 I=1,IZON
C3100          WRITE(21,*)XPZO(I),YPRI(I),IZOP(I),ITZO(I)
C3200     6400 CONTINUE
C3300          CLOSE(UNIT=21,FILE=NAME2,DISPOSE='SAVE')
C3400          WRITE(ITTY,6500)NAME2
C3500     6500 FORMAT(4X,'(ABOVE INFO HAS BEEN WRITTEN TO FILE ',A10,')')
C3600     6666  RETURN
C3700          END
C3800          SUBROUTINE MATIN(XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
C3900    COMMENT
C4000    C
C4100    C       SUBROUTINE MATIN FOR READING THE MATRIX INFORMATION
C4200    C
C4300    COMMENT
C4400          DIMENSION IZOP(21),ITZO(21),YPRI(21),XPZO(21)
C4500          COMMON/COM1/IDUN,K1,TRAD,IPLN,ITTX,ITTY,CONT
C4600          DOUBLE PRECISION NAME1
C4700          WRITE(ITTY,10)
C4800     10   FORMAT(/,4X,'NAME OF FILE FOR MATRICE ? ',$)
C4900          READ(ITTX,20)NAME1
C5000     20   FORMAT(A10)
C5100          OPEN(UNIT=21,FILE=NAME1,ACCESS='SEQIN')
C5200          READ(21,*)IZON,ITOT,K1
C5300          DO 30 I=1,IZON
C5400          READ(21,*)XPZO(I),YPRI(I),IZOP(I),ITZO(I)
C5500     30   CONTINUE
C5600          CLOSE(UNIT=21,FILE=NAME1,DISPOSE='SAVE')
C5700     6666 RETURN
C5800          END
```

```
C                    ******* SUB24.FOR ********
C
C                    SUBROUTINE PLOT
C
      SUBROUTINE PLOT1(IPLO,XARR,YARR,IZ01,IDEC,ZARX,ZARY,IZ02,TITL,NWO,
     1 NFO,TIT2,NW2,NF2,IEX)
COMMENT
C        SUBROUTINE PLOT
C                                    XARR AND YARR OF LENGTH IZ01 FOR POINTS
C                                    ZARX AND ZARY OF LENGTH IZ02 FOR LINE
C                                       IN OVERLAY (PREDICTED VALUES)
C                                    (MAX LENGHT OF ARRAYES 250)
C                                    IPLO  1 : NORMAL
C                                    IPLO  2 : OVERLAY
C                                    IDEC  1 : LINE
C                                    IDEC  2 : POINTS
C                                    TITL    : CHARACTER ARRAY WITH
C                                    NWO     : NO OF WORDS OF
C                                    NFO     : FORM OF WORDS
C                                    IEX     : 1 ALSO VDU PLOT ; 2 NOT
C
C                                    SEE DIMENSION FOR ARRAYS
COMMENT
      DIMENSION XPLO(250),YPLO(250),XARR(IZ01),YARR(IZ01),ZARX(IZ02),
     1 ZARY(IZ02),XAR1(250),YAR1(250),TITL(NWO),TIT2(NW2),
     2 ZPLX(250),ZPLY(250),TIM(3) ,IP(250),IST1(250)
      LOGICAL OVLA
      COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTY,CONT
      IF (IDUM.NE.1) GO TO 5555
      OVLA=.FALSE.
      IF (IPLO.EQ.2) OVLA=.TRUE.
C
C     SORT THE ARRAYS AFTER XARR
C
      IFAI=1
      CALL M01AAF(XARR,1,IZ01,IP,IST1,IFAI)
      CALL IFTEST(IFAI,1,ITTY)
      IF (IFAI.NE.0) GO TO 6666
      DO 10 I=1,IZ01
      XAR1(IP(I))=XARR(I)
      YAR1(IP(I))=YARR(I)
   10 CONTINUE
C
C     FINDS MAX AND MIN
C
      XMIN=0.0
      YMIN=0.0
      ZMIN=0.0
      VMIN=0.0
      XMAX=0.0
      YMAX=0.0
      ZMAX=0.0
      VMAX=0.0
      DO 4450 I=1,IZ01
      XMIN=AMIN1(XMIN,XAR1(I))
      XMAX=AMAX1(XMAX,XAR1(I))
      YMAX=AMAX1(YMAX,YAR1(I))
      YMIN=AMIN1(YMIN,YAR1(I))
 4450 CONTINUE
      IF (XMIN.GT.0.0) XMIN=0.0
```

```
06100           IF (.NOT.OVLA) GO TO 4465
06200           DO 4460 I=1,IZ02
06300           ZMAX=AMAX1(ZMAX,ZARY(I))
06400           ZMIN=AMIN1(ZMIN,ZARY(I))
06500           VMAX=AMAX1(VMAX,ZARX(I))
06600           VMIN=AMIN1(VMIN,ZARX(I))
06700    4460 CONTINUE
06800    4465 CONTINUE
06900       C
07000       C   INITIAL INFORMATION ON SIZE OF PLOT
07100       C
07200           DATA XMID,YHE1,XRIO,YRIO/200.0,180.0,35.0,35.0/
07300           XTMA=AMAX1(XMAX,VMAX)
07400           XTMI=AMIN1(XMIN,VMIN)
07500           XDIF=XTMA-XTMI
07600       C
07700       C    CALCULATES THE CONVERSION FACTORS FOR X AND Y
07800       C         FROM MATHEMATICAL SCALE TO MM.
07900       C
08000           XCON=(XMID-XRIO)*0.80/XDIF
08100           YTMA=AMAX1(YMAX,ZMAX)
08200           YTMI=AMIN1(YMIN,ZMIN)
08300           YDIF=YTMA-YTMI
08400           YCON=(YHE1-YRIO)*0.80/YDIF
08500       C
08600       C   CALCULATES THE NEW COORDINATES IN MM
08700       C
08800           DO 4500 I=1,IZ01
08900           XPLO(I)=((XAR1(I)-XTMI)*XCON)+XRIO
09000           YPLO(I)=((YAR1(I)-YTMI)*YCON)+YRIO
09100    4500 CONTINUE
09200           IF (.NOT.OVLA) GO TO 4475
09300           DO 4470 I=1,IZ02
09400           ZPLX(I)=((ZARX(I)-XTMI)*XCON)+XRIO
09500           ZPLY(I)=((ZARY(I)-YTMI)*YCON)+YRIO
09600    4470 CONTINUE
09700    4475 CONTINUE
09800       C
09900       C   DRAWING THE FRAME
10000       C
10100           CALL TIME(TIM)
10200           DATA TIM(2)/'*.'/
10300           CALL CC563S
10400           CALL ERRMAX(50)
10500           CALL DEVPAP(XMID,XHE1,1)
10600           CALL MOVTO2(0.0,0.0)
10700           CALL LINTO2(XMID,0.0)
10800           CALL LINTO2(XMID,XHE1)
10900           CALL LINTO2(0.0,XHE1)
11000           CALL LINTO2(0.0,0.0)
11100       C
11200       C       IF IDEC 1 PLOTS THE LINE THROUGH POINTS OF XARR YARR
11300       C
11400           IF (IDEC.EQ.2) GO TO 5500
11500           CALL MOVTO2(XPLO(1),YPLO(1))
11600           DO 5000 I=2,IZ01
11700           CALL LINTO2(XPLO(I),YPLO(I))
11800    5000 CONTINUE
11900           GO TO 5600
12000       C
```

```
C      IF IDEC 2 PLOTS THE POINTS OF XARR YARR
C
 5500 CONTINUE
      CALL SYMTO2(XPLO,YPLO,IZ01,8)
 5600 CONTINUE
      IF (.NOT.OVLA) GO TO 5010
C
C       PLOTS THE LINE THROUGH POINTS ZARX ZARY
C
      CALL MOVTO2(ZPLX(1),ZPLY(1))
      DO 5700 I=2,IZ02
      CALL LINTO2(ZPLX(I),ZPLY(I))
 5700 CONTINUE
 5010 CONTINUE
C
C    AXIS TYPE SCALE AND POSITION
C
        CALL AXIPOS(1,XRIO,YRIO,(0.80*(XWID-XRIO)),1)
        CALL AXIPOS(1,XRIO,YRIO,(0.80*(YHE1-YRIO)),2)
        CALL AXISCA(3,10,XTMI,XTMA,1)
        CALL AXISCA(3,10,YTMI,YTMA,2)
        CALL AXIDRA(-1,1,1)
        CALL AXIDRA(1,-1,2)
C
C     TEXT OUTPUT ETC.
C
        CALL MOVTO2(70.0,160.0)
        CALL CHAARR(TITL,NWO,NFO)
        CALL MOVTO2(69.0,159.0)
        IF (OVLA) CALL LINTO2(121.0,159.0)
        IF (.NOT.OVLA) CALL LINTO2(133.0,159.0)
        IF (.NOT.OVLA) GO TO 5020
        CALL MOVTO2(80.0,140.0)
        CALL CHAARR(TIT2,NW2,NF2)
        CALL MOVTO2(55.0,10.0)
        CALL CHAHOL(23HNO. OF VISITORGROUPS/*.)
        CALL CHAINT(K1,10)
        GO TO 5030
 5020 CONTINUE
        CALL MOVTO2(55.0,10.0)
        CALL CHAARR(TIT2,NW2,NF2)
 5030 CONTINUE
        CALL MOVTO2(1.5,1.5)
C       CALL CHAARR(TIM,2,5)
        CALL CHAHAR(1,0)
        CALL CHAHOL(10HPLOT NO:*.)
        CALL CHAINT(IPLN,-3)
        CALL MOVTO2(15.0,70.0)
        CALL CHAHAR(0,1)
        CALL CHAHOL(28HWILLINGNESS TO PAY (PENCE)*.)
        CALL DEVEND
        WRITE(ITTY,5100)IPLN
 5100 FORMAT(4X,'( GINO-F PLOT NO :',I3,')',/)
        IPLN=IPLN+1
        IF (IEX.NE.1) GO TO 5555
C
C     OUTPUT ON VDU
C
        CALL ITY
        CALL DEVPAP(160.0,120.0,1)
```

```
18100           CALL AXISCA(3,20,0.0,XMAX,1)
18200           CALL AXISCA(3,20,0.0,YMAX,2)
18300           CALL GRAF(XAR1,YAR1,IZ01,0)
18400           CALL DEVEND
18500      5555 CONTINUE
18600      6666 RETURN
18700           END
```

```
C                    ******** SUB35.FOR **********
C
C                    SUBROUTINES TRNF GLM2  LSFI2  WEIGHT HEADIN
C                                FUN1 FUN2
C
C
C
C
      SUBROUTINE TRLF(IWEI,XARR,YARR,INOB,XWEI)
COMMENT
C          SUBROUTINE TRFN   LN(Y)=LN(A)-B*LN(X)
C                                XARR AND YARR OF LENGTH INOB
C                                IWEI : 1 NORMAL TRFN
C                                       2 - 7 WEIGHTED
C
C                                SEE DIMENSION STATEMENT FOR ARRAYS
C                  ESTIMATED PARAMETERS IN COMMON VAR1/ XRES(20)
C
COMMENT
      DIMENSION XARR(INOB),YARR(INOB),TITL(10),
     1  TIT2(10),TIT3(10),XAR1(21),YAR1(21),XWEI(INOB),XAR2(21),
     2  YAR2(21),XPRP(100),YPRP(100)
      COMMON/DUM1/IDUM,K1,TRAD,IPLN,IITX,ITTY,CONT
      COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
      EXTERNAL FUN2
      CONT=.TRUE.
C
C     OUTPUTS HEADING
C
      WRITE(ITTY,10)
   10  FORMAT(1H1,//,4X,'TRANSFORMED LIN. REG.:',/,4X,'**********
     1****',/)
      IF (IWEI.NE.1) CALL HEADIN(IWEI,ITTY)
      CALL PRESS(3,XARR,YARR,INOB,XWEI,XPRES)
C   TO PREVENT EXTREMES WHEN TRANSFORMING 1.0 IS ADDED
      DO 15 I=1,INOB
      XAR2(I)=ALOG(XARR(I)+1.0)
      YAR2(I)=ALOG(YARR(I)+1.0)
   15  CONTINUE
      IFAIL=1
      CALL SORT1(XARR,YARR,INOB,XAR1,YAR1,IFAIL,ITTY)
      IF (IFAIL.NE.0) CONT=.FALSE.
      IF (IFAIL.NE.0) GO TO 6666
C
C     CALL TO SUBROUTINE GLMMY
C
      CALL GLMMY(INOB,XAR2,YAR2,XWEI)
C
C     CALCULATES SS-RES FOR ORIGINAL POINTS ON BASIS OF WEIGHTED
C                ANALYSIS PARAMETERS
C
      SSRES=0.0
        DO 20 I=1,INOB
   20  SSRES=SSRES+((YARR(I)-FUN2(XARR(I)))**2)
   30  CONTINUE
C
C      POINTS FOR PREDICTED LINE
C
      IF (IDUM.NE.1) GO TO 200
      XDIF=XAR1(INOB)-XAR1(1)
      XINT=XDIF/99
```

```
      35    CONTINUE
            DO 40 I=1,100
            XPRP(I)=XAR1(1)+(XINT*(I-1))
            YPRP(I)=FUN2(XPRP(I))
            INO=I
      C
      C     STOPS IF YPRP NEGATIVE  (MEANS NEGATIVE VISITORS)
      C
            IF (YPRP(I).LT.0.0) GO TO 100
      40    CONTINUE
            GO TO 200
      C
      C     FINDS THE INTERSECT OF YAXIS AND CURVE
      C              (STOPS DRAWING NEGATIVE VISITORGROUPS)
      100   CONTINUE
            XMAX=XPRP(INO)*1.1
            IFAI=1
            EPS=0.0005
            ETA=0.005
      C
      C     CALLS  NAG-SUBROUTINE
      C
            CALL C05ACF(0.0,XMAX,EPS,ETA,FUN2,XSTA,IFAI)
            CALL IFIEST(IFAI,2,ITTY)
            IF (IFAI.NE.0) CONT=.FALSE.
            IF (IFAI.NE.0) GO TO 6666
            XINT=(XSTA-XAR1(1))/99.0
            GO TO 35
      200   CONTINUE
      C
      C     OUTPUT RESULT
      C
            WRITE(ITTY,50)XRES(8),XRES(9)
      50    FORMAT(4X,'F-VALUE  =',F10.5,/,4X,'R-SQUARE =',4X,F6.5)
            WRITE(ITTY,65)XRES(6)
      65    FORMAT(4X,'SS-RES   =',E15.6)
            WRITE(ITTY,80)SSRES
      80    FORMAT(4X,'SS-RES BASED ON ORIGINAL POINTS =',E15.6)
            WRITE(ITTY,75)XPRES
      75    FORMAT(4X,'PRESS-value =',F15.2,/)
            WRITE(ITTY,60)XRES(1),XRES(2)
      60    FORMAT(4X,'REGRESSION CONSTANT:    A =',E15.5,/,4X,
           1          'REGRESSION COEFFICIENT: B =',E15.5,/)
            WRITE(ITTY,70)
      70    FORMAT(6X,'( LN(Y) = A - B*LN(X) )',/)
      C
      C     PLOTS THE TRIP DEMAND CURVE
      C
            DATA TIT2/'LINE FROM TRNF PROCEDURE*.'/
            DATA TIT3/'LINE FROM WEIGHTED TRNF PROCEDURE*.'/
            DATA TITL/'TRIP DEMAND CURVE*.'/
            IF (IWEI.EQ.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRP,XPRP,100,
           1           TITL,4,5,TIT2,5,5,2)
            IF (IWEI.NE.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRP,XPRP,100,
           1          TITL,4,5,TIT3,7,5,2)
      6666  RETURN
            END
            SUBROUTINE LSFI2(IWEI,XARR,YARR,INOB,XWEI)
      COMMENT
      C
```

```
C  SUBROUTINE LSFI2      2'DG POLYNOMIAL
C
C                    SEE GLM2 FOR PARAMETERS OF CALL
C                 EST. PARAMETERS STORED IN COMMON VAR1/ XRES2
C
      COMMENT
            DIMENSION XARR(INOB),YARR(INOB),XWEI(INOB),TITL(10),YAR1(21),
           1 XPRP(100),YPRP(100),TIT2(10),TIT3(10),YPRE(21),
           2 XAR1(21),XAR2(21)
            COMMON /VAR1/XRES(20),XRES2(20),XC2F(3)
            COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT
            EXTERNAL FUN1
            CONT=.TRUE.
            IFAI1=1
C
C     OUTPUTS HEADING
C
            WRITE(ITTY,4380)
 4380 FORMAT(1H1,//,4X,'2ND DG POLYNOMIAL FIT:',/,4X,'*****************
           1****',/)
            IF (IWEI.NE.1) CALL HEADIN(IWEI,ITTY)
            CALL PRESS(2,XARR,YARR,INOB,XWEI,XPRES)
            CALL SORT1(XARR,YARR,INOB,XAR1,YAR1,IFAI1,ITTY)
            IF (IFAI1.NE.0) CONT=.FALSE.
            IF (IFAI1.NE.0) GO TO 6666
            DO 400 I=1,INOB
  400 XAR2(I)=(XARR(I)**2)
C
C     CALL TO GLMMY2
C
            CALL GLMMY2(INOB,XARR,XAR2,YARR,XWEI)
C
C     CALCULATES PREDICTED VALUES
C
            IF (IDUM.NE.1) GO TO 200
            XDIF=XAR1(INOB)-XAR1(1)
            XINT=XDIF/99
            K=0
   35       CONTINUE
            DO 40 I=1,100
            XPRP(I)=XAR1(1)+(XINT*(I-1))
            YPRP(I)=XRES2(1)+XRES2(2)*XPRP(I)+XRES2(3)*(XPRP(I)**2)
            INO=I
C
C     STOPS IF YPRP NEGATIVE (MEANS NEGATIVE VISITORS)
C
            IF (YPRP(I).LT.0.0) GO TO 100
   40       CONTINUE
            GO TO 200
C
C     FINDS THE INTERSECT OF YAXIS AND CURVE
C              (STOPS DRAWING NEGATIVE VISITORGROUPS)
  100       CONTINUE
            XMAX=XPRP(INO)*1.1
            IFAI=1
            EPS=0.0005
            ETA=0.005
C
C     CALLS NAG-SUBROUTINE
C
```

```
      CALL COSALF(0.0,XMAX,EPS,ETA,FUN1,XSTA,IFAI)
      CALL IFIEST(IFAI,2,ITTY)
      IF (IFAI..F.0) CONT=.FALSE.
      IF (IFAI..E.0) GO TO 6666
      XINT=(XSTA-XAR1(1))/99.0
      K=K+1
      IF (K.LT.50) GO TO 35
      WRITE(ITTY,180)
  180 FORMAT(/,4X,'STOP ON K IN SUBR LSFI2',/)
  200 CONTINUE
C
C   CALCULATES SS-RES FOR  POINTS
C
      SSRES=0.0
      DO 300 I=1,INOB
      YPRE(I)=FUN1(XARR(I))
      SSRES=((YARR(I)-YPRE(I))**2)+SSRES
  300 CONTINUE
C
C     OUTPUTS RESULTS
C
       WRITE(ITTY,75)XRES2(6)
   75  FORMAT(4X,'SS-RES    =',E15.6)
       IF (INEI.NE.1) WRITE(ITTY,80)SSRES
   80  FORMAT(4X,'SS-RES BASED ON ORIGINAL POINTS =',E15.6)
       WRITE(ITTY,90)XPRES
   90  FORMAT(4X,'PRESS-value =',F15.2,/)
       WRITE(ITTY,60)XRES2(1),XRES2(2),XRES2(3)
   60  FORMAT(4X,'COEFFICIENTS ARE :',3X,'COEF(1) ',E13.5,/,25X,
      1'COEF(2) ',E13.5,/,25X,'COEF(3) ',E13.5,/,
      2  /,6X,'( Y = COEF(1) + COEF(2)*X + COEF(3)*X^2 )',/)
C
C     PLOTS THE TRIP DEMAND CURVE
C
      DATA TITL/'TRIP DEMAND CURVE*.'/
      DATA TIT2/'2ND DG POLYNOMIAL CURVEFIT*.'/
      DATA TIT3/'2ND DG POLYNOMIAL WEIGHTED CURVEFIT*.'/
      IF (INEI.EQ.1) CALL PLOT1(2,YAR1,XAR1,INOB,2,YPRP,XPRP,100,
     1   TITL,4,5,TIT2,6,5,2)
      IF (INEI.NE.1) CALL PLOT1(2,YAR1,XAR1,INOB,2,YPRP,XPRP,100,
     1 TITL,4,5,TIT3,8,5,2)
 6666 RETURN
      END
      SUBROUTINE HEADIN(INEI,ITTY)
COMMENT
C
C    HEADINGS FOR GLM2 LSFI2 EXP2
C
COMMENT
      INTEGER INEI,ITTY
      IF (INEI.EQ.2) WRITE(ITTY,4200)
 4200 FORMAT(4X,'( WEIGHTED ACCORDING TO NO. OF VISITORGROUPS )',/)
      IF (INEI.EQ.3) WRITE(ITTY,4400)
 4400 FORMAT(4X,'( WEIGHTED ACCORDING TO SQRT(ZONEPOPULATION) )',/)
      IF (INEI.EQ.4) WRITE(ITTY,4600)
 4600 FORMAT(4X,'( WEIGHTED ACCORDING TO SQRT(ZONEPOPULATION) ',/,
     1       20X,' * NO  OF VISITORGROUPS FROM ZONE )',/)
      IF (INEI.EQ.5) WRITE(ITTY,4800)
 4800 FORMAT(4X,'( WEIGHTED ACCORDING TO ZONEPOP./VISITRATE ) ',/)
      IF (INEI.EQ.6) WRITE(ITTY,5000)
```

```
 3900      5000 FORMAT(4X,'( SUBJECTIVE WEIGHTING ) ',/)
 4000           IF (IWEI.EQ.7) WRITE(ITTY,5200)
 4100      5200 FORMAT(4X,'( WEIGHTED ACCORDING TO ZONEPOPULATION ) ',/)
 4200      6666 RETURN
 4300           END
 4400           SUBROUTINE WEIGHT(INO1,XPZO,ITZO,IZOP)
 4500      COMMENT
 4600      C
 4700      C      CREATES THE WEIGHING ARRAYS
 4800      C
 4900      COMMENT
 5000           INTEGER INO1,ITZO(INO1),IZOP(INO1)
 5100           REAL XPZO(INO1)
 5200           COMMON/WEIA/WEI1(21),WEI2(21),WEI3(21),WEI4(21),WEI5(21),WEI6(21)
 5300          1 ,WEI7(21)
 5400           DO 100 I=1,INO1
 5500           WEI1(I)=1.0
 5600           WEI2(I)=SQRT(FLOAT(IZOP(I)))
 5650           IF (IZOP(I).EQ.0)   WEI2(I)=1.0
 5700           WEI3(I)=FLOAT(IZOP(I))/XPZO(I)
 5750           IF (XPZO(I).EQ.0.0)  WEI3(I)=1.0
 5775           IF (IZOP(I).EQ.0)  WEI3(I)=1.0
 5800           WEI4(I)=FLOAT(ITZO(I))
 5900           WEI5(I)=WEI2(I)*WEI4(I)
 6000           WEI6(I)=FLOAT(IZOP(I))
 6050           IF (IZOP(I).EQ.0)   WEI6(I)=1.0
 6100      100  CONTINUE
 6200      6666 RETURN
 6300           END
 6400           REAL FUNCTION FUN1(X)
 6500      COMMENT
 6600      C
 6700      C    FUNCTION   FUN1   2ND OG POLYNOMIAL
 6800      C
 6900      C                    EST. PARAMETERS IN COMMON VAR1/ XRES2
 7000      COMMENT
 7100           REAL X
 7200           COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
 7300           FUN1=XRES2(1)+XRES2(2)*X+XRES2(3)*(X**2)
 7400           RETURN
 7500           END
 7600           REAL FUNCTION FUN2(X)
 7700      COMMENT
 7800      C
 7901      C    FUNCTION   FUN2   TRANSFORMED LINEAR REG
 8000      C
 8100      C                    EST. PARAMETERS IN COMMON VAR1/ XRES
 8200      COMMENT
 8300           REAL X
 8400           COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
 8500           FUN2=EXP(XRES(1)+(XRES(2)*ALOG(X+1.0)))-1.0
 8600           RETURN
 8700           END
 8800           SUBROUTINE GLM2(INEI,XARR,YARR,INOB,XMEI)
 8900      COMMENT
 9000      C          SUBROUTINE GLM2
 9100      C                                   XARR AND YARR OF LENGTH INOB
 9200      C                                   INEI : 1 NORMAL GLM
 9300      C                                          2 - 7 WEIGHTED
 9400      C
```

```
C                                        SEE DIMENSION STATEMENT FOR ARRAYS
C                              ESTIMATED PARAMETERS IN COMMON VAR1/ XRES(20)
C
      COMMENT
            DIMENSION XARR(INOB),YARR(INOB),XPRE(2),YPRE(2),TITL(10),
           1  TIT2(10),TIT3(10),XAR1(21),YAR1(21),XWEI(INOB)
            COMMON/DUM1/IDUM,K1,TRAD,IPLM,ITTX,ITTY,CONT
            COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
            CONT=.TRUE.
C
C     OUTPUTS HEADING
C
            WRITE(ITTY,10)
   10       FORMAT(1H1,//,4X,'LINEAR REGRESSION FIT:',/,4X,'******************
           1****',/)
            IF (IWEI.NE.1) CALL HEADIN(IWEI,ITTY)
            CALL PRESS(1,XARR,YARR,INOB,XWEI,XPRES)
            IFAI1=1
            CALL SORT1(XARR,YARR,INOB,XAR1,YAR1,IFAI1,ITTY)
            IF (IFAI1.NE.0) CONT=.FALSE.
            IF (IFAI1.NE.0) GO TO 6666
C
C     CALL TO SUBROUTINE GLMMY
C
            CALL GLMMY(INOB,XARR,YARR,XWEI)
            IF (IWFI.EQ.1) GO TO 30
C
C     CALCULATES SS-RES FOR ORIGINAL POINTS ON BASIS OF WEIGHTED
C                 ANALYSIS PARAMETERS
C
            SSRES=0.0
              DO 20 I=1,INOB
   20       SSRES=SSRES+((YARR(I)-(XRES(1)+XPES(2)*XARR(I)))**2)
   30       CONTINUE
C
C      POINTS FOR PREDICTED LINE
C
            IF (IDUM.NE.1) GO TO 200
            XPRE(1)=0.0
            YPRE(2)=0.0
            YPPE(1)=XRES(1)
            XPPE(2)=-XPES(1)/XRES(2)
C
C     OUTPUT RESULT
C
  200       WRITE(ITTY,50)XRES(8),XRES(9)
   50       FORMAT(4X,'F-VALUE  =',F10.5,/,4X,'R-SQUARE =',4X,F6.5)
             WRITE(ITTY,75)XRES(6)
   75       FORMAT(4X,'SS-RES   =',E15.6)
             IF (IWEI.NE.1) WRITE(ITTY,80)SSRES
   80       FORMAT(4X,'SS-RES BASED ON ORIGINAL POINTS =',E15.6)
            WRITE(ITTY,90)XPRES
   90       FORMAT(4X,'PRESS-value =',F15.2,/)
            WRITE(ITTY,60)XRES(1),XRES(2)
   60       FORMAT(4X,'REGPESSION CONSTANT:     A =',E15.5,/,4X,
           1          'REGRESSION COEFFICIENT: B =',E15.5,/)
            WRITE(ITTY,70)
   70       FORMAT(6X,'( Y = A + B*X )',/)
C
C      PLOTS THE TRIP DEMAND CURVE
```

```
C
      DATA IIT2/'LINE FROM GLM PROCEDURE*.'/
      DATA IIT3/'LINE FROM WEIGHTED GLM PROCEDURE*.'/
      DATA IITL/'TYPP DEMAND CURVE*.'/
      IF (IWEI.EQ.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRE,XPRE,2,
     1          IITL,4,5,IIT2,5,5,2)
      IF (IWEI.NE.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRE,XPRE,2,
     1          IITL,4,5,IIT3,7,5,2)
 6666 RETURN
      END
```

```
C                    ******** SUB45.FOR *********
C
C                    SUBROUTINES REDUC MAXMI AGGR2 SORT1 ISTEST
C
C
      SUBROUTINE REDUC(ITEX,XARR,YARR,IZON,IZOP,ITZO,ITOT)
COMMENT
C
C     SUBROUTINE REDUCES XARR,YARR FOR ZERO ELEMENTS
C
C                                   XARR AND YARR OF LENGTH IZON
C                                   IZOP : ZONEPOPULATION
C                                   ITZO : NO OF  VISITORGR. FROM ZONE
C                                    (IF ITZO=0 OBSERVATION IS EXCLUDED)
C                                   ITEX : 1 TEXT ; 2 NOT
C                                   ITTY : OUTPUT CHANNEL
C
C                        SEE DIMENSION STATEMENT FOR ARRAYS
C
COMMENT
      DIMENSION XARR(IZON),YARR(IZON),IZOP(IZON),XAR1(21),YAR1(21),IZO1(
     121),ITZO(IZON),ITZ1(21)
      COMMON/DUM1/IDUM,X1,TRAD,IPLN,ITTX,ITTY,CONT
      K=0
      DO 100 I=1,IZON
      IF (ITZO(I).EQ.0) GO TO 50
      K=K+1
      XAR1(K)=XARR(I)
      YAR1(K)=YARR(I)
      IZO1(K)=IZOP(I)
      ITZ1(K)=ITZO(I)
   50 CONTINUE
  100 CONTINUE
      IOUT=IZON-K
      IZON=K
      ITOT=0
      DO 200 I=1,IZON
      XARR(I)=XAR1(I)
      YARR(I)=YAR1(I)
      IZOP(I)=IZO1(I)
      ITZO(I)=ITZ1(I)
      ITOT=ITOT+ITZO(I)
  200 CONTINUE
      IF ((IOUT.EQ.0).OR.(ITEX.NE.1)) GO TO 300
        WRITE(ITTY,220)
  220   FORMAT(/,4X,'***** WARNING *******',/)
        WRITE(ITTY,250)IOUT
  250 FORMAT(/,4X,I4,' OBSERVATION(S) EXCLUDED DUE TO ZERO VISITORS',/
     1)
  300 CONTINUE
      RETURN
      END
      SUBROUTINE AGGR2(ITYP,IAN1,XARR,YARR,IZON,IZOP,
     1 ITZO,ITOT)
COMMENT
C
C   SUBROUTINE CALCULATING AGGREGATED DEMAND FROM TRIP DEMAND
C
C   ITYP : 1 BASED ON ZONEPOINTS
C        : 2 BASED ON INDIVIDUALS !!!!
```

```
C          : S REGULATES TO ACTUAL OBSERVED
C                              XARR AND YARR OF LENGTH IZON
C                              IAN1 : 1 GLM ; 2 POLYNOMIAL ; 3 TRNF
C                              IZOP : ZONEPOPULATION
C                              ITZO : NO. OF VISITORGROJPS FROM ZONE
C                              ITOT : TOTAL NO OF OBSERVATIONS INCLUDED IN AN.
C                              ITTX : INPUT CHANNEL
C                              ITTY : OUTPUT CHANNEL
C
C                                   SEE DIMENSION STATEMENT FOR ARRAYS
C
C                                   PARAMETERS ARE STORED IN COMMON VAR1/
C
      COMMENT
            DIMENSION XAPR(IZON),YARR(IZON),IZOP(IZON),YPRE(21),XNUB(200),
           1XAGG(250),YAGG(250),TITL(10),TIT2(10),ITZO(IZON)
            EXTERNAL FUN1,FUN2
            COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
            COMMON/DUM1/IDUM,A1,TRAD,IPLN,ITTX,ITTY,CONT
            IF (.NOT.CONT) GOTO 6666
C
C              INITIALIZE ARRAYS
C
            DO 10 I=1,250
            XAGG(I)=0.0
            YAGG(I)=0.0
      10    CONTINUE
            DO 15 J=1,200
            XNUB(J)=0.0
      15    CONTINUE
             CALL MAXMI(XARR,YARR,IZON,XMAX,YMAX,XMIN,YMIN)
            ISIM=0
            YINC=5.0
            EPS=0.00005
            ETA=0.005
            YINI=(YMAX-YMIN)/10
            K=0
C
C             FINDS YMAX FOR INTEGRATION (INTERSECT)
C
      25    CONTINUE
            IFAI=1
            K=K+1
            GOTO (50,50,40)IAN1
      50    YMAX2=YMIN+(YINI*K)
              CALL COSACF(0.0,YMAX2,EPS,ETA,FUN1,YSTA,IFAI)
              IF ((IFAI.NE.0).AND.(K.LT.25)) GO TO 25
              CALL IFIEST(IFAI,4,ITTY)
              XCM=-XRES2(2)/(XRES2(3)*2)
              IF ((FUN1(XCM).GT.0.0).AND.(XRES2(3).GE.0.0)) GO TO 35
              GO TO 60
      35      CONTINUE
C             SUPPLOPTION FOR POLYNIMIAL - INTEGRATES TO YMAX
              WRITE(ITTY,36)XCM
      36      FORMAT(/,4X,'***** WARNING *******',/,4X,'NO INTERSECT - WILL IN
           1TEGRATE TO VALUE FOR WHICH',/,4X,'FUNCTION HAS MINIMUM - EQUAL TO'
           2,F8.2,' PENCE ',/)
              YSTA=XCM
              GO TO 60
      40       YSTA= EXP(-XRES(1)/XRES(2))
```

```
12100             GO TO 60
12200      50     YSTA=-XRES(1)/XRES(2)
12300      60    YLMA=YSTA
12400      C
12500      C      TEST OF SIZE OF Y INCREMENT
12600      C
12700             CALL ISIZE(YLMA,YINC,IIIY)
12800             IF (IIYP.EQ.2) REWIND 22
12900             IF (IIYP.NE.2) ICOUN=IZON
13000             IF (IIYP.EQ.2) ICOUN=ITOT
13100             DO 100 I=1,ICOUN
13200             IF (IIYP.EQ.2) READ(22,46) YVAR
13300      46     FORMAT(F10.2)
13400             IF (IIYP.NE.2) YVAR=YARR(I)
13500              ISTP=INT((YLMA-YVAR)/YINC)+1
13600             IF (YLMA.LE.YVAR) GO TO 100
13700             ISTM=MAX0(ISTM,ISTP)
13800      C
13900      C      CALCULATES NO OF VISITORGROUPS FOR GIVEN FEE
14000      C
14100               DO 70 J=1,ISTP
14200               YINT=YVAR+((J-1)*YINC)
14300               IF (YINT.GE.YLMA) GO TO 100
14400               IF (IAN1.EQ.2) XSTA=FUN1(YINT)
14500               IF (IAN1.EQ.1) XSTA=XRES(1)+XRES(2)*YINT
14600               IF (IAN1.EQ.3) XSTA=FUN2(YINT)
14700               IF (J.EQ.1) XPAR=XSTA
14800               IF (IIYP.EQ.1) XNUB(J)=(XSTA*IZOP(I)/K1)+XNUB(J)
14900               IF (IIYP.EQ.2) XNUB(J)=(XSTA/XPAR)+XNUB(J)
15000               IF (IIYP.EQ.3) XNUB(J)=(XSTA*IIZO(I)/XPAR)+XNUB(J)
15100      70     CONTINUE
15200     100     CONTINUE
15300             DO 600 J=1,ISTM
15400             YAGG(J)=(YINC*(J-1))
15500             XAGG(J)=XAGG(J)+XNUB(J)
15600     600     CONTINUE
15700      C
15800      C      PLOTS THE AGGREGATED DEMAND CURVE
15900      C
16000             DATA IIIL/'AGGREGATED DEMAND CURVE*.'/
16100             DATA IIT2/'NO. OF VISITORGROUPS*.'/
16200             CALL PLOT1(1,XAGG,YAGG,ISTM,1,XAGG,YAGG,ISTM,IIIL,6,5,IIT2,5,5,1)
16300      C
16400      C SIMPLE INTEGRATION
16500      C
16600             XCON=0.0
16700             DO 700 I=2,ISTM
16800             XCON=XCON+((XAGG(I-1)-XAGG(I))*YINC*0.5)+(XAGG(I)*YINC)
16900      700    CONTINUE
17000             XCON=XCON/100.0
17100      C
17200      C      OUTPUT RESULT
17300      C
17400             WRITE(IITY,750)XCON
17500      750    FORMAT(/,4X,'CONSUMERS SURPLUS EQUALS',15X,F16.2,' POUNDS')
17600             GOTO(900,1000,1100)IIYP
17700      900    WRITE(IITY,800)ITOT
17800      800    FORMAT(/,4X,'ACTUAL OBSERVED NO OF VISITORGROUPS',I5)
17900             GOTO 5555
18000     1000    WRITE(IITY,754)
```

```
      754  FORMAT(4X,'( BASED ON INDIVIDUAL INTEGRATION )',/)
           GOTO 5555
      1100 WRITE(ITTY,752)
      752   FORMAT(4X,'( REGULATED TO OBSERVED NO OF VISITORGROUPS )',/)
      5555 WRITE(ITTY,802)XAGG(1)
      802  FORMAT(4X, 'PREDICTED FROM THE MODEL          ',F8.0,/)
      6664 RETURN
           END
           SUBROUTINE SORT1(XARR,YARR,INOB,XAR4,YAR4,IFAI,ITTY)
      COMMENT
      C
      C    SUBROUTINE SORT1  (XARR=ARRAY TO BE SORTED AFTER)
      C                       XAR4 AND YAR4 ARE THE SORTED OUTPUT ARRAYS
      C
      COMMENT
           DIMENSION XARR(INOB),YARR(INOB),XAR4(INOB),YAR4(INOB),IP(4000),
          1 IST1(4000)
           IFAI=1
           CALL MOIAAF(XARR,1,INOB,IP,IST1,IFAI)
             CALL IFIFST(IFAI,1,ITTY)
             IF (IFAI.NE.0) GO TO 6666
           DO 10 I=1,INOB
           XAR4(IP(I))=XARR(I)
           YAR4(IP(I))=YARR(I)
       10  CONTINUE
       6666 RETURN
           END
           SUBROUTINE ISTEST(YLMA,YINC,ITTY)
      COMMENT
      C
      C     SUBROUTINE CHECKING NO OF STEPS IN INTEGRATION
      C
      C                              YLMA : YMAX FOR INTEGRATION
      C                                     (TRIP DEMAND CURVES INTERSECT WITH
      C                                      Y-AXIS )
      C                              YINC : INCREMENT FOR INTEGRATION
      C                              ITTY : OUTPUT CHANNEL
      C
      COMMENT
           ISTP=INT(YLMA/YINC)
           IF (ISTP.LT.200) GO TO 500
           YINC2=YLMA/180.0
           WRITE(ITTY,100)YINC,YINC2
       100 FORMAT(/,4X,'INCREMENT IN Y FOR CALCULATING AGGREGATED DEMAND',
          1   /,4X,'HAS BEEN CHANGED FROM ',10X,F3.0,' TO ',F3.0,' PENCE',/)
           YINC=YINC2
       500 CONTINUE
           RETURN
           END
           SUBROUTINE RANNAM(ITYP,FNAME)
      COMMENT
      C
      C    GENERATING PSEUDONAME FOR TMP-FILES
      C
      COMMENT
           DOUBLE PRECISION FNAME
           INTEGER IR(5),ITYP
           INTEGER G05DYF
      C
      C   SETS THE RANDOM GENERATOR TO A NONREPEATABLE STATE
```

```
C
      CALL GO5CCF
      DO 10 I=1,5
C
C   GIVES INTEGER RANDOM NUMBERS IN THE INTERVAL 1,9
C
   10 IR(I)=GO5CYF(1,9)
C
C  ENCODE NOT STANDARD FORTRAN
C
      IF (IITYP.EQ.1)  ENCODE(10,100,FNAME)IR
      IF (IITYP.EQ.2)  ENCODE(10,200,FNAME)IR
  100 FORMAT(1HC,5I1,4H.TMP)
  200 FORMAT(1HC,5I1,4H.MAT)
 6666 RETURN
      END
      SUBROUTINE MAXMI(XARR,YARR,IZON,XMAX,YMAX,XMIN,YMIN)
COMMENT
C
C   SUBROUTINE MAXMI  TO FIND MAX AND MIN OF THE TWO ARRAYS
C                     XARR AND YARR OF LENGTH IZON
C
COMMENT
      DIMENSION XARR(IZON),YARR(IZON)
      XMIN=0.0
      YMIN=0.0
      XMAX=0.0
      YMAX=0.0
      DO 100 I=1,IZON
      XMAX=AMAX1(XMAX,XARR(I))
      YMAX=AMAX1(YMAX,YARR(I))
      XMIN=AMIN1(XMIN,XARR(I))
      YMIN=AMIN1(YMIN,YARR(I))
  100 CONTINUE
      RETURN
      END
```

```
00100      C                ******* SUB55.FOR **********
00200      C
00300      C                SUBROUTINES IFTEST SUBWEI GLMMY GLMMY2 PRESS
00400      C
00500      C
00600            SUBROUTINE IFIEST(IFAIL,IDENT,ITTY)
00700      COMMENT
00800      C
00900      C       SUBROUTINE CHECKING THE IFAI VALUES FROM NAG LIBRARY
01000      C
01100      *COMMENT
01200            IF (IFAIL.EQ.0) GO TO 400
01300            IF (IDENT.EQ.1) WRITE(ITTY,50)IFAIL
01400            IF (IDENT.EQ.2) WRITE(ITTY,100)IFAIL
01500            IF (IDENT.EQ.3) WRITE(ITTY,150)IFAIL
01600            IF (IDENT.EQ.4) WRITE(ITTY,200)IFAIL
01700       50   FORMAT(/,4X,'FAILURE IN SUBROUTINE CALL TO M01AAF; IFAIL=',
01800           1 2X,I2,/,4X,'PROGRAMME WILL ADVANCE TO NEXT PROCEDURE !',/)
01900      100   FORMAT(/,4X,'FAILURE IN SUBROUTINE CALL TO C05ACF  ;',9X,' IFAIL='
02000           1 ,2X,I2,/,4X,'PROGRAMME WILL ADVANCE TO NEXT PROCEDURE !',/)
02100       150  FORMAT(/,4X,'FAILURE IN SUBROUTINE CALL TO G02CAF; IFAIL=',
02200           1 2X,I2,/,4X,'PROGRAMME WILL ADVANCE TO NEXT PROCEDURE !',/)
02300       200  FORMAT(/,4X,'FAILURE IN SUBROUTINE CALL C05CAF :',11X,' IFAIL =',
02400           1 2X,I2,/,4X,'PROBAPLY NO INTERSECT BETWEEN TRIP DEMAND CURVE ',
02500           2   'AND Y-AXIS !!',/)
02600       400  CONTINUE
02700      6666  RETURN
02800            END
02900            SUBROUTINE SUBWEI(IN01,XWEI,SUBW)
03000      COMMENT
03100      C
03200      C
03300      C       SUBJECTIVE WEIGHTED POINTS
03400      C
03500      COMMENT
03600            DIMENSION XWEI(IN01)
03700            COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT
03800            WRITE(ITTY,10)
03900       10   FORMAT(/,4X,'SUBJECTIVE WEIGHTING ',/)
04000            IF (.NOT.SUBW) GOTO 100
04100            GOTO 30
04200       20   SKIP RECORD ITTX
04300       30   WRITE(ITTY,40)
04400       40   FORMAT(/,4X,'THE SAME WEIGHTING AS BEFORE (YES OR NO) ? ',$)
04500            READ(ITTX,50,ERR=20)ANSW
04600       50   FORMAT(A3)
04700            IF (ANSW.NE.'YES') GO TO 100
04800            WRITE(ITTY,60)
04900       60   FORMAT(/,4X,'WEIGHTING USED:',/,4X,'****************',/)
05000            DO 80 I=1,IN01
05100            WRITE(ITTY,70)I,XWEI(I)
05200       70   FORMAT(6X,'ZONE NO. ',I2,' ; WEIGHT ',E15.8)
05300       80   CONTINUE
05400            GO TO 5555
05500      100   DO 200 I=1,IN01
05600      110   GO TO  130
05700      120   SKIP RECORD ITTX
05800      130   WRITE(ITTY,140)I
05900      140   FORMAT(/,6X,'ZONE NO. ',I2,' WEIGHT ? ',$)
06000            READ(ITTX,*,ERR=120)XWEI(I)
```

```
06100           IF (BATCH) WRITE(ITTY,150)XWEI(I)
06200      150  FORMAT(E15.8)
06300           XTOT=XTOT+XWEI(I)
06400      200  CONTINUE
06500      5555 SUBW=.TRUE.
06600      6666 RETURN
06700           END
06800           SUBROUTINE GLMWY(IOBS,XAR,YAR,WEI)
06900      COMMENT
07000      C
07100      C      WEIGHTED GLM PROCEDURE
07200      C
07300      C          XAR YAR ARRAYES OF LENGTH IOBS
07400      C          WEI : ARRAY FOR WEIGHTS
07500      COMMENT
07600           INTEGER IOBS
07700           REAL XAR(21),YAR(21),WEI(21),YPRE(21)
07800           DOUBLE PRECISION SX,SY,SXX,SYY,SXY,ACONS,BCOEF,DF,
07900          2 SSTOT,SSRES,SSREG,MSREG,MSRES,FVAL,RSQ,XMEAN,YMEAN,STRES
08000           COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
08100      C
08200      C    INITIATING ARRAYS
08300      C
08400             DF=0.0
08500             SX=0.0
08600             SY=0.0
08700             SXX=0.0
08800             SYY=0.0
08900             SXY=0.0
09000      C
09100      C CALCULATING SS'
09200      C
09300             DO 300 I=1,IOBS
09400             DF=DF+DBLE(WFI(I))
09500             SX=SX+DBLE(XAR(I))*DBLE(WEI(I))
09600             SY=SY+DBLE(YAR(I))*DBLE(WEI(I))
09700             SXX=SXX+(DBLE(XAR(I))**2)*DBLE(WEI(I))
09800             SYY=SYY+(DBLE(YAR(I))**2)*DBLE(WEI(I))
09900             SXY=SXY+DBLE(XAR(I))*DBLE(YAR(I))*DBLE(WEI(I))
10000       300   CONTINUE
10100             BCOEF=(SXY-(SX*SY/DF))/(SXX-((SX**2)/DF))
10200             ACONS=SY/DF-BCOEF*SX/DF
10300             SSTOT=SYY-SY*SY/DF
10400             SSREG=ACONS*SY+BCOEF*SXY-SY*SY/DF
10500             MSREG=SSREG/1.0
10600             SSRES=SSTOT-SSREG
10700             MSRES=SSRES/(DF-2.0)
10800             FVAL=MSREG/MSRES
10900             RSQ=SSREG/SSTOT
11000             XMEAN=SX/DF
11100             YMEAN=SY/DF
11200      C
11300      C      STORING RESULTS IN ARRAY XRES
11400      C
11500             XRES(1)=SNGL(ACONS)
11600             XRES(2)=SNGL(BCOEF)
11700             XRES(3)=0.0
11800             XRES(4)=SNGL(SSREG)
11900             XRES(5)=SNGL(MSREG)
12000             XRES(6)=SNGL(SSRES)
```

```
12100            XRES(7)=SNGL(MSRES)
12200            XRES(8)=SNGL(FVAL)
12300            XRES(9)=SNGL(RSQ)
12400            XRES(10)=SNGL(XMEAN)
12500            XRES(11)=SNGL(YMEAN)
12600            XRES(12)=SNGL(DF)
12700            XRES(13)=SNGL(SSTOT)
12800     6666 RETURN
12900          END
13000        SUBROUTINE GLMMY2(IOBS,XAR1,XAR2,YARR,WEI)
13100    COMMENT
13200    C
13300    C      WEIGHTED GLM PROCEDURE  WITH 2 INDEPENDANT VARIABLES
13400    C
13500    C                                XAR1 XAR2 YARR ARRAYES OF LENGHT IOBS
13600    C                                WEI :ARRAY FOR WEIGHTS
13700    C
13800    COMMENT
13900          INTEGER IOBS
14000          REAL XAR1(IOBS),XAR2(IOBS),YARR(IOBS),WEI(IOBS)
14100           DOUBLE PRECISION  E11,E21,E31,E12,E22,E32,E13,E23,E33,DET,
14200         1 SX1,SX2,SX1X1,SX1X2,SX2X2,SY,SYY,SX1Y,SX2Y,B1,B2,B3,DF,SQDT,
14300         2 SSTOT,SSRES,SSREG,MSREG,MSRES,FVAL,RSQ,XMEAN,YMEAN,STRES
14400          COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
14500    C
14600    C    INITIATING ARRAYS
14700    C
14800            DF=0.0
14900            Sx1=0.0
15000            Sx2=0.0
15100            SY=0.0
15200            Sx1x2=0.0
15300            Sx1x1=0.0
15400            Sx2x2=0.0
15500            SYY=0.0
15600            Sx1Y=0.0
15700            Sx2Y=0.0
15800    C
15900    C CALCULATING SS'
16000    C
16100            DO 100 I=1,IOBS
16200            DF=DF+DBLE(WEI(I))
16300            SX1=SX1+DBLE(XAR1(I))*DBLE(WEI(I))
16400            SX2=SX2+DBLE(XAR2(I))*DBLE(WEI(I))
16500            SY=SY+DBLE(YARR(I))*DBLE(WEI(I))
16600            SX1X1=SX1X1+(DBLE(XAR1(I))**2)*DBLE(WEI(I))
16700            SX2X2=SX2X2+(DBLE(XAR2(I))**2)*DBLE(WEI(I))
16800            SX1X2=SX1X2+(DBLE(XAR1(I))*DBLE(XAR2(I))*DBLE(WEI(I)))
16900            SX1Y=SX1Y+(DBLE(XAR1(I))*DBLE(YARR(I))*DBLE(WEI(I)))
17000            SX2Y=SX2Y+(DBLE(XAR2(I))*DBLE(YARR(I))*DBLE(WEI(I)))
17100            SYY=SYY+(DBLE(YARR(I))**2)*DBLE(WEI(I))
17200     100    CONTINUE
17300    C
17400    C     DETERMINANT
17500    C
17600           DET=DF*SX1X1*SX2X2+SX1*SX1X2*SX2*2-SX2*SX1X1*SX2-
17700         1   SX1*SX1*SX2X2-DF*SX1X2*SX1X2
17800            SQDT=DSQRT(DET)
17900    C
18000    C     INVERTING THE X'X MATRICE
```

```
18100     C
18200     C       E11= (SX1X1*SX2X2-SX1X2*SX1X2)/DET
18300               E11= (SX1X1/SQDT)*(SX2X2/SQDT)-(SX1X2/SQDT)*(SX1X2/SQDT)
18400     C       E21= -(SX1*SX2X2-SX1X2*SX2)/DET
18500               E21= -((SX1/SQDT)*(SX2X2/SQDT)-(SX1X2/SQDT)*(SX2/SQDT))
18600     C       E31= (SX1*SX1X2-SX2*SX1X1)/DET
18700               E31= (SX1/SQDT)*(SX1X2/SQDT)-(SX2/SQDT)*(SX1X1/SQDT)
18800     C       E12= -(SX1*SX2X2-SX1X2*SX2)/DET
18900               E12= -((SX1/SQDT)*(SX2X2/SQDT)-(SX1X2/SQDT)*(SX2/SQDT))
19000     C       E22= (DF*SX2X2-SX2*SX2)/DET
19100               E22= (DF/SQDT)*(SX2X2/SQDT)-(SX2/SQDT)*(SX2/SQDT)
19200     C       E32= -(DF*SX1X2-SX1*SX2)/DET
19300               E32= -((DF/SQDT)*(SX1X2/SQDT)-(SX1/SQDT)*(SX2/SQDT))
19400     C       E13= (SX1*SX1X2-SX1X1*SX2)/DET
19500               E13= (SX1/SQDT)*(SX1X2/SQDT)-(SX1X1/SQDT)*(SX2/SQDT)
19600     C       E23= -(DF*SX1X2-SX1*SX2)/DET
19700               E23= -((DF/SQDT)*(SX1X2/SQDT)-(SX1/SQDT)*(SX2/SQDT))
19800     C       E33= (DF*SX1X1-SX1*SX1)/DET
19900               E33= (DF/SQDT)*(SX1X1/SQDT)-(SX1/SQDT)*(SX1/SQDT)
20000     C
20100     C    CALCULAIING B1 B2 B3
20200     C
20300               B1= E11*SY+E12*SX1Y+E13*SX2Y
20400               B2= E21*SY+E22*SX1Y+E23*SX2Y
20500               B3= E31*SY+E32*SX1Y+E33*SX2Y
20600     C
20700     C    CALCULAIING SSTOT SSRES ETC
20800     C
20900               SSTOT=SYY-SY*(SY/DF)
21000               SSREG=B1*SY+B2*SX1Y+B3*SX2Y-SY*(SY/DF)
21100               MSREG=SSREG/2.0
21200               SSRES=SSTOT-SSREG
21300               MSRES=SSRES/(DF-3.0)
21400               FVAL=MSREG/MSRES
21500               RSQ=SSREG/SSTOT
21600               XMEAN=SX/DF
21700               YMEAN=SY/DF
21800     C
21900     C      STORING RESULTS IN ARRAY XRES2
22000     C
22100               XRES2(1)=SNGL(B1)
22200               XRES2(2)=SNGL(B2)
22300               XRES2(3)=SNGL(B3)
22400               XRES2(4)=SNGL(SSREG)
22500               XRES2(5)=SNGL(MSREG)
22600               XRES2(6)=SNGL(SSRES)
22700               XRES2(7)=SNGL(MSRES)
22800               XRES2(8)=SNGL(FVAL)
22900               XRES2(9)=SNGL(RSQ)
23000               XRES2(10)=SNGL(XMEAN)
23100               XRES2(11)=SNGL(YMEAN)
23200               XRES2(12)=SNGL(DF)
23300               XRES2(13)=SNGL(SSTOT).
23400      6666   RETURN
23500             END
23600             SUBROUTINE PRESS(IAN,X,Y,INOB,XWEI,XPRES)
23700     COMMENT
23800     C
23900     C   SUBROUTINE PRESS - CALCULATES PRESS VALUES
24000     C                      IAN: 1 GLM2
```

```
C                                   2 LSFI2
C                                   3 TRFN
C                               X AN Y : ARRAYS OF LENGTH INOB
C                               INOB: NO OF OBSERVATIONS
C                               XWEI: ARRAY OF LENGTH INOB WITH WEIGTHS
C                               XPRES: RETURNS PRESS VALUE
C
COMMENT
      DIMENSION X(INOB),Y(INOB),XNEW(21),YNEW(21)
     1 ,X2NEW(21),XLOG(21),YLOG(21),XWEI(INOB)
      COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT/VAR1/XRES(20),
     1  XRES2(20),XC2F(3)
      EXTERNAL FUN2
       XPRES=0.0
       DO 1000 I=1,INOB
       K=0
       DO 500 J=1,INOB
       IF (J.EQ.I) GO TO 500
       K=K+1
       XNEW(K)=X(J)
       X2NEW(K)=X(J)**2
       YNEW(K)=Y(J)
  500  CONTINUE
       INEW=INOB-1
COMMENT FUNCTION TO TEST
      GOTO (520,540,560)IAN
  520 CALL GLSMY(INEW,XNEW,YNEW,XWEI)
      YPRE=XRES(1)+XRES(2)*X(I)
      GO TO 560
  540 CALL GLSMY2(INEW,XNEW,X2NEW,YNEW,XWEI)
      YPRE=XRES2(1)+XRES2(2)*X(I)+XRES2(3)*(X(I)**2)
      GO TO 580
  560 DO 565 ICOUN=1,INEW
      XLOG(ICOUN)=ALOG(XNEW(ICOUN)+1.0)
  565 YLOG(ICOUN)=ALOG(YNEW(ICOUN)+1.0)
      CALL GLSMY(INEW,XLOG,YLOG,XWEI)
      YPRE=FUN2(X(I))
      GO TO 580
  580 CONTINUE
      XADD=(Y(I)-YPRE)**2
      XPRES=XPRES+XADD
 1000 CONTINUE
       RETURN
      END
```

Appendix A5.2

Computer Programme PFAFO1.FOR

40

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

```
FFFFF   333    4  4     999                 J  BBBB     CCCC  H   H
    3     3   4  4    9   9                 J  B   B   C     H   H
          3   4  4    9   9                 J  B   B   C     H   H
FFF       3   44444   9999                  J  BBBB    C     HHHHH
          3       4       9           J     J  B   B   C     H   H
    3     3       4       9           J     J  B   B   C     H   H
     333          4    999             JJJ    BBBB     CCCC  H   H
```

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

START User F349 JBCH [146,24155] Job PFAF01 Seq. 3698 Date 10-Mar-83 23:52:0

```
00000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000111111111122222222223333333333444444444455555555556666666666677777777788
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
00100          PROGRAM PFAF01
00200     COMMENT READS DATA FROM PSOF.AA ; FOREST NAME POSTCODES & NO OF VISITOR
00300     C          GROUPS FROM POSTZONE
00400     C        FINDS ZONEPOPULATION FROM PSOF.BB
00500     C        FINDS DISTANCE FROM PSOF.CC
00600     C        LOGICAL DISTANCE FROM PSOF.DD
00700     C        EXTRA DISTANCES (ZERO OBS.) FROM PSOF.EE
00800     C          FINDS MAX DISTANCE (COST) FOR OBSERVED VISITS
00900     C          FINDS THEN THE MISSING POSTZONES AND ADDS THESE OBSERVATIONS
01000     C                AS POINTS (DIST,0.0)
01100     C          WRITES TO RESULTFILE PFAF01.RES :
01200     C              FOREST NAME  ETC. , NO OF OBSERVATIONS
01300     C              AND FOR EACH OBSERVATION :
01400     C              VISITRATE, COST, NO OF VISITORGROUPS, ZONEPOPULATION
01500     C
01600     C      TEST RESULTS TO FILE FOR31.DAT  AND ERRORS TO FOR32.DAT
01700     C
01800          DIMENSION FNAM2(7),XDIS2(100,2)
01900          COMMON/COD0/FNAME(7),IZOP(200),INOV(200),IZNO(200),COUNT(1),
02000         . IFORN,ASTAY,LOG1(200),LOG2(200),CFONO
02100          COMMON/COD1/IPOP(250,2)
02200          COMMON/COD2/IX(200),IY(200)
02300          COMMON/COD3/XDIS(60,100,2)
02400          COMMON/COD4/XALT(250,3)
02500          COMMON/COD5/JR,JS,JORI,XPCT(200),YCOS(200)
02600          TEST=.TRUE.  !   IF TEST TRUE TESTFILE = FOR31.DAT
02700          CALL PCOIN(TEST)
02800          CALL DISIN(TEST)
02900          CALL NEARN(TEST)
03000          CALL MPLOT ; INU=0 ! INITIALISE PLOTROUTINE
03100          OPEN(UNIT=21,FILE='PSOF.AA',ACCESS='SEQIN')  ! NO OF VISITORS
03200          OPEN(UNIT=23,FILE='PFAF01.RES') ! RESULTFILE
03300          OPEN(UNIT=24,FILE='PSOF.MAT') ! TO BE USED WITH CLAW12.EXE
03400          OPEN(UNIT=20,FILE='PSOF.EE',ACCESS='SEQIN') ! ZONES WITH ZERO VIS.
03500          JR=0
03600          DO 3 I=1,200
03700          LOG1(I)=.FALSE.
03800        3 LOG2(I)=.FALSE.
03900        5 READ(21,10)START
04000       10 FORMAT(A5)
04100          IF (START.NE.'START') GO TO 5 ! CONTINUE TO READ UNTIL FIRST START
04200       15 READ(21,20,END=6666)COUNT(1),CFONO,ASTAT,IFORN,(FNAME(I),I=1,7)
04300       20 FORMAT(A5,X,I2,X,F7.5,I3,7A5)
04400          J=0
04500       25 READ(21,30)NEXT
04600       30 FORMAT(A4)
04700          IF (NEXT.EQ.'NEXT') GO TO 50 ! NEXT INDICATES A NEW FOREST
04800          BACKSPACE 21  ! REREADS THE RECORD IF STILL THE SAME FOREST
04900          READ(21,40)(IX(I),IY(I),I=J+1,J+8)
05000       40 FORMAT(8(I4,I5,X))
05100          J=J+8
05200          GO TO 25
05300     C  END OF FIRST FOREST  FROM PSOF.AA
05400       50 CONTINUE
05500     C  WRITE STATEMENTS FOR CHECK OF FILE PSOF.AA
05600          IF (.NOT.TEST) GO TO 55
05700          WRITE(31,53)
05800          WRITE(31,20)COUNT(1),CFONO,ASTAT,IFORN,(FNAME(I),I=1,7)
05900       53 FORMAT(4X,'CHECK OF FILE PSOF.AA')
06000          DO 52 I=1,J
```

```
06100        52 WRITE(31,54)IX(I),IY(I)
06200        54 FORMAT(4X,I4,4X,I8)
06300     55    JS=0;IMISS=0;XMDIS=0.0;ITOT=0
06400           IF (J.EQ.0) GO TO 45  ! IF NO OBSERVATIONS IN FOREST
06500           CALL PCAGG(J,JR)  ! IF MISSING VALUES THEN IN ARRAY ELEMENT JR
06600           JS=JR
06700           IF (IX(JR).NE.999) GO TO 58 ! SKIPS THE MISSING VALUES
06800           IMISS=IY(JR)
06900           JS=JR-1
07000           JORI=JS
07100     58    DO 100 I=1,JS
07200           CALL POPUS(IX(I),IPOPU,$100)
07300           CALL DISTA(IFORN,IX(I),XDIST,$100)
07400           IF (XDIST.GT.XMDIS) XMDIS=XDIST ! FINDS MAX OBSERVED DIST
07500           CALL NELOG(IX(I),XDIST,LOG1(I),LOG2(I))
07600           XPCT(I)=FLOAT(IY(I))/FLOAT(IPOPU)  ! CHANGE TO REAL !!!!!
07700           FACT=1.0 ! FACT HAS TO BE CHANGED
07800           YCOS(I)=XDIST*FACT*2.0  ! FACT IS KM-PRICE
07900           IZOP(I)=IPOPU
08000           INOV(I)=IY(I)
08100    100    ITOT=ITOT+IY(I)
08200           REWIND 20
08300    210    READ(20,220,ERR=210,END=300)(FNAM2(I),I=1,7)
08400    220    FORMAT(19X,7A5)
08500           DO 230 I=1,7
08600    230    IF (FNAM2(I).NE.FNAME(I)) GO TO 210 ! CHECKS ITS THE FOREST
08700           JC=0
08800           DO 232 I=1,100
08900           DO 232 J=1,2
09000       232 XDIS2(I,J)=0.0
09100       235 READ(20,240,END=300)NEXT
09200       240 FORMAT(A4)
09300           IF (NEXT.EQ.'NEXT') GO TO 260
09400           BACKSPACE 20
09500           READ(20,250)(XDIS2(I,1),XDIS2(I,2),I=JC+1,JC+8)
09600       250 FORMAT(8(F4.0,F5.1,X))
09700           JC=JC+8
09800           GO TO 235
09900       260 IF (JC.EQ.0) GO TO 300
10000     C   WRITE STATEMENT FOR CHECK OF FILE PSOF.EE
10100           IF (.NOT.TEST) GO TO 268
10200           WRITE(31,262)
10300       262 FORMAT(4X,'CHECK OF FILE PSOF.EE ')
10400           DO 264 I=1,JC
10500       264 WRITE(31,266)XDIS2(I,1),XDIS2(I,2)
10600       266 FORMAT(4X,F5.0,3X,F6.2)
10700       268 CONTINUE
10800           DO 300 I=1,JC
10900           IF ((XDIS2(I,2).GT.XMDIS).OR.(XDIS2(I,2).EQ.0.0)) GO TO 300
11000           JS=JS+1
11100           IX(JS)=INT(XDIS2(I,1))
11200           XPCT(JS)=0.0
11300           YCOS(JS)=XDIS2(I,2)*FACT*2.0
11400           CALL NELOG(IX(JS),XDIS2(I,2),LOG1(JS),LOG2(JS))
11500           CALL POPUS(IX(JS),IPOPU,$300)
11600           IZOP(JS)=IPOPU
11700           INOV(JS)=0
11800           IY(JS)=0
11900       300 CONTINUE
12000     C  OUTPUT TO FOR34.DAT OF MAX OBSERVED DISTANCE
```

```
2100       45  WRITE(34,98)(FNAME(J),J=1,7),XMDIS
2200       98  FORMAT(4X,7A5,4X,F8.2,' KM.')
2300           WRITE(23,60)COUNT(1),CFONO,(FNAME(I),I=1,7),IFORN,ASTAT,JS,ITOT
2400          . ,IMISS,XMDIS
2500       60  FORMAT(//,4X,A5,I3,3X,7A5,/,4X,I4,4X,F9.5,3X,I3,2(2X,I8),2X,F6.2,
2600          . /)
2700           IF (J.EQ.0) GO TO 15 ! NO OBS WRITES ONLY HEADING
2800           DO 400 I=1,JS
2900           WRITE(23,70)IX(I),IY(I),IZOP(I),XPCT(I),YCOS(I),LOG1(I),LOG2(I)
3000       70  FORMAT(4X,I5,2X,I6,2X,I8,2X,F15.8,2X,F15.8,2(2X,L4))
3100       400 CONTINUE
3200       C       GO TO 80
3300       C       IF (IFORN.GT.4) GO TO 80  ! BECAUSE OF TEST
3400           INU=INU+1
3500           CALL PLOT2(INU,FNAME)
3600           GO TO 80  ! BECAUSE OF TEST
3700           K1=1.0  ! K1 FACTOR FOR RATE OF VISITS  - TO BE CHANGED
3800           WRITE(24,*)JR,ITOT,K1,IFORN ! MATRIX FOR CLAW12.EXE
3900           DO 150 I=1,JR
4000        150 WRITE(24,*)XPCT(I),YCOS(I),IZOP(I),INOV(I)
4100         80 GO TO 15
4200       6666 CONTINUE  ! JUMP TO HERE WHEN END OF FILE
4300           CALL PLOTND
4400           STOP
4500           END
4600       C
4700       C    PCOIN
4800       C                WITH ENTRY POPUS
4900       C
5000           SUBROUTINE PCOIN(TEST)
5100       C  READS THE POSTCODES AND POPULATIONS INTO AN ARRAY  IPOP IN COMMON
5200           COMMON/COD1/IPOP(250,2)
5300           OPEN(UNIT=22,FILE='PSOF.BB',ACCESS='SEQIN')
5400           J=0
5500        5  READ(22,10,END=30)(IPOP(J+I,1),IPOP(J+I,2),I=1,7)
5600       10  FORMAT(7(I4,X,I5,X))
5700           J=J+7
5800           GO TO 5
5900       30  CLOSE(UNIT=22,FILE='PSOF.BB',DISPOSE='SAVE')
6000       C       CHECK OF FILE PSOF.BB
6100           IF (.NOT.TEST) GO TO 36
6200           WRITE(31,33)
6300           DO 35 I=1,J
6400       35  WRITE(31,32)IPOP(I,1),IPOP(I,2)
6500       32  FORMAT(4X,I4,4X,I8)
6600       33  FORMAT(4X,'CHECK OF FILE PSOF.BB ')
6700       36  RETURN
6800       C
6900       C  ENTRY POPUS
7000       C
7100           ENTRY POPUS(ISEAR,IPOPU,*)
7200       C  FINDS THE POPULATION FOR GIVEN POSTCODE FROM ARRAY IPOP IN COMMON
7300           DO 100 I=1,250
7400           J=I
7500        100 IF (IPOP(I,1).EQ.ISEAR) GO TO 200
7600           RETURN 1    ! NO SUCCESS :RETURNS TO LINENUMBER IN CALL
7700        200 IPOPU=IPOP(J,2) ! IPOPU RETURNS VALUE
7800           RETURN  ! RETURNS TO CALLING LINE
7900           END
8000       C
```

```
18100    C    DISIN
18200    C       WITH ENTRY DISTA
18300    C
18400            SUBROUTINE DISIN(TEST)
18500    C  READS THE POSTCODE DISTANCE FILE INTO AN ARRAY IDIS IN COMMON
18600            DIMENSION FNAME(7)
18700            COMMON/COD3/XDIS(60,100,2)
18800            OPEN(UNIT=23,FILE='PSOF.CC',ACCESS='SEQIN')
18900            DO 4 I=1,60
19000            DO 4 J=1,100
19100            DO 4 K=1,2
19200        4   XDIS(I,J,K)=0.0
19300        5   READ(23,15,END=200)IFORN,(FNAME(I),I=1,7)  ! READS THE FOREST NO
19400       15   FORMAT(9X,I2,7X,7A5)     ! AND FOREST NAME
19500            J=0
19600      100   READ(23,110,END=200)NEXT
19700      110   FORMAT(A4)
19800            IF (NEXT.EQ.'NEXT') GO TO 150
19900            BACKSPACE 23  ! REREADS THE RECORD WHEN STILL SAME FOREST
20000            READ(23,120)(XDIS(IFORN,I,1),XDIS(IFORN,I,2),I=J+1,J+8)
20100      120   FORMAT(8(F4.0,F5.1,X))
20200            J=J+8
20300            GO TO 100
20400      150   CONTINUE
20500    C    TO CHECK FILE  PSOF.CC
20600            IF (.NOT.TEST) GO TO 180
20700            WRITE(31,163)
20800            WRITE(31,155)(FNAME(I),I=1,7),IFORN
20900      155   FORMAT(/,4X,7A5,2X,I3)
21000      163   FORMAT(4X,'CHECK OF FILE PSOF.CC ')
21100            DO 160 I=1,J
21200      160   WRITE(31,170)XDIS(IFORN,I,1),XDIS(IFORN,I,2)
21300      170   FORMAT(4X,F5.0,2X,F6.2)
21400      180   GO TO 5
21500      200   CLOSE(UNIT=23,FILE='PSOF.CC',DISPOSE='SAVE')
21600            RETURN
21700    C
21800    C  ENTRY DISTA
21900    C
22000            ENTRY DISTA(IF2RN,IPOCO,XDIST,*)
22100    C  FINDS FOR GIVEN FOREST DISTANCE TO GIVEN POSTCODE USING ARRAY XDIS
22200            DO 300 I=1,100
22300            J=I
22400      300   IF (INT(XDIS(IF2RN,I,1)).EQ.IPOCO) GO TO 400
22500            RETURN 1 ! NO SUCCESS : RETURNS TO LINENUMBER IN CALL
22600      400   XDIST=XDIS(IF2RN,J,2)
22700            RETURN ! SUCCESS : RETURNS TO CALLING LINE
22800            END
22900    C
23000    C      PCCHA
23100    C
23200            SUBROUTINE PCCHA(IPOCO,IPOCK,*)
23300    C    SUBROUTINE PCCHA  TO CHANGE POSTCODES INSIDE COUNTY OF COPENHAGEN
23400    C    AND CORRECT FOR MISSING VALUES IN THE WHOLE REGION
23500            COMMON/COD0/FNAME(7),IZOP(200),INOV(200),IZNO(200),COUNT(1),
23600           . IFORN,ASTAY,LOG1(200),LOG2(200),CFONO
23700            IPOCK=999
23800            IF ((IPOCO.GE.4700).OR.(IPOCO.LE.999)) GO TO 50
23900            IF (IPOCO.LT.1500) IPOCK=1000
24000            IF ((IPOCO.GE.1500).AND.(IPOCO.LT.2000)) IPOCK=1500
```

```
4100          IF ((IPOCO.GE.2000).AND.(IPOCO.LT.2100)) IPOCK=2000
4200          IF ((IPOCO.GE.2100).AND.(IPOCO.LT.2200)) IPOCK=2100
4300          IF ((IPOCO.GE.2200).AND.(IPOCO.LT.2300)) IPOCK=2200
4400          IF ((IPOCO.GE.2300).AND.(IPOCO.LT.2400)) IPOCK=2300
4500          IF ((IPOCO.GE.2400).AND.(IPOCO.LT.2450)) IPOCK=2400
4600          IF ((IPOCO.GE.2450).AND.(IPOCO.LT.2500)) IPOCK=2450
4700          IF (IPOCO.EQ.4060) GO TO 50
4800          IF ((IPOCO.GE.2500).AND.(IPOCO.LT.4100)) GO TO 100
4900          IF (IPOCO.EQ.4621) GO TO 100
5000          IF (IPOCU.EQ.4622) GO TO 100
5100     50   RETURN  !     ; RETURNS TO CALLING LINE
5200    100   CALL PDPUS(IPOCD,IPOPU,$75) ! CHECK POPULATION AND DISTANCE
5300          CALL DISTA(IFORN,IPOCO,XDIST,$85) ! IN INT 2500 TO 4100
5400          IPOCK=IPOCO     !   - IF NOT EXIST, GO TO 50 AND RETURN
5500          RETURN 1 !            IPOCK=999 -  OTHERWISE TO LINENUM IN CALL
5600    C  TO CHECK IF ALL POPULATIONS EXIST
5700      75  WRITE(32,80)IPOCO
5800      80  FORMAT(4X,'POPULATION DOES NOT EXIST FOR POSTZONE ',I5,' !! ')
5900          GO TO 50
6000      85  WRITE(32,90)IFORN,IPOCO
6100      90  FORMAT(4X,'NO DISTANCE FOUND FOR FOREST NO.',I3,' TO POSTCODE NO.'
6200         . ,I5,' !! ')
6300          GO TO 50
6400          END
6500    C
6600    C     PCAGG
6700    C
6800          SUBROUTINE PCAGG(J,JR)
6900          DIMENSION IX2(200),IY2(200)
7000          COMMON/COD2/IX(200),IY(200)
7100          J0999=0;J1000=0;J1500=0;J2000=0;J2100=0;J2200=0;J2300=0;J2400=0;
7200          J2450=0
7300          L0999=.FALSE.;L1000=.FALSE.;L1500=.FALSE.;L2000=.FALSE.;L2100=.FAL
7400         .SE.;L2200=.FALSE.;L2300=.FALSE.;L2400=.FALSE.;L2450=.FALSE.
7500          K=0
7600          DO 5000 I=1,J
7700          IF (IY(I).EQ.0) GO TO 5000
7800          CALL PCCHA(IX(I),IPOCK,$5000)
7900     999  IF (IPOCK.NE.999) GO TO 1000
8000          J0999=J0999+IY(I)
8100          L0999=.TRUE.
8200    1000  IF (IPOCK.NE.1000) GO TO 1500
8300          J1000=J1000+IY(I)
8400          L1000=.TRUE.
8500    1500  IF (IPOCK.NE.1500) GO TO 2000
8600          J1500=J1500+IY(I)
8700          L1500=.TRUE.
8800    2000  IF (IPOCK.NE.2000) GO TO 2100
8900          J2000=J2000+IY(I)
9000          L2000=.TRUE.
9100    2100  IF (IPOCK.NE.2100) GO TO 2200
9200          J2100=J2100+IY(I)
9300          L2100=.TRUE.
9400    2200  IF (IPOCK.NE.2200) GO TO 2300
9500          J2200=J2200+IY(I)
9600          L2200=.TRUE.
9700    2300  IF (IPOCK.NE.2300) GO TO 2400
9800          J2300=J2300+IY(I)
9900          L2300=.TRUE.
10000   2400  IF (IPOCK.NE.2400) GO TO 2450
```

```
30100           J2400=J2400+IY(I)
30200           L2400=.TRUE.
30300      2450 IF (IPOCK.NE.2450) GO TO 5000
30400           J2450=J2450+IY(I)
30500           L2450=.TRUE.
30600      5000  CONTINUE
30700           L=0
30800           IF (.NOT.L0999) GO TO 10
30900           L=L+1
31000           IX2(L)= 999
31100           IY2(L)=J0999
31200        10 IF (.NOT.L1000) GO TO 20
31300           L=L+1
31400           IX2(L)=1000
31500           IY2(L)=J1000
31600        20 IF (.NOT.L1500) GO TO 30
31700           L=L+1
31800           IX2(L)=1500
31900           IY2(L)=J1500
32000        30 IF (.NOT.L2000) GO TO 40
32100           L=L+1
32200           IX2(L)=2000
32300           IY2(L)=J2000
32400        40 IF (.NOT.L2100) GO TO 50
32500           L=L+1
32600           IX2(L)=2100
32700           IY2(L)=J2100
32800        50 IF (.NOT.L2200) GO TO 60
32900           L=L+1
33000           IX2(L)=2200
33100           IY2(L)=J2200
33200        60 IF (.NOT.L2300) GO TO 70
33300           L=L+1
33400           IX2(L)=2300
33500           IY2(L)=J2300
33600        70 IF (.NOT.L2400) GO TO 80
33700           L=L+1
33800           IX2(L)=2400
33900           IY2(L)=J2400
34000        80 IF (.NOT.L2450) GO TO 90
34100           L=L+1
34200           IX2(L)=2450
34300           IY2(L)=J2450
34400        90 DO 200 I=1,J
34500           IF (IY(I).EQ.0) GO TO 200
34600           CALL PCCHA(IX(I),IPOCK,s100)
34700           GO TO 200
34800       100 L=L+1
34900           IX2(L)=IPOCK
35000           IY2(L)=IY(I)
35100       200 CONTINUE
35200           JR=L
35300           IF (L0999) GO TO 400
35400           DO 300 I=1,JR
35500           IX(I)=IX2(I)
35600     C     WRITE(5,*)IX(I),IY2(I)
35700       300 IY(I)=IY2(I)
35800           GO TO 6666
35900       400 DO 500 I=2,JR
36000           IX(I-1)=IX2(I)
```

```
  500 IY(I-1)=IY2(I)
      IX(JR)=IX2(1)
      IY(JR)=IY2(1)
 6666 RETURN
      END
C
C     NEARN
C          WITH ENTRY NELOG
C
      SUBROUTINE NEARN(TEST)
C  ESTABLISH MATRIX FOR LOGICAL NEAREST FOREST
      COMMON/COD4/XALT(250,3)
      OPEN(UNIT=22,FILE='PSOF.DD',ACCESS='SEQIN')
      J=0
   5  READ(22,10,END=30)(XALT(I+J,1),XALT(I+J,2),XALT(I+J,3),I=1,7)
  10  FORMAT(7(F4.0,2(F3.0),X))
      J=J+7
      GO TO 5
  30  CLOSE(UNIT=22,FILE='PSOF.DD',DISPOSE='SAVE')
C    CHECK OF FILE PSOF.DD
      IF (.NOT.TEST) GO TO 37
      WRITE(31,33)
      DO 35 I=1,J
  35  WRITE(31,32)(XALT(I,K),K=1,3)
  33  FORMAT(4X,'CHECK OF FILE PSOF.DD ')
  32  FORMAT(4X,F5.0,2(2X,F3.0))
  37  RETURN
C
C    ENTRY NELOG
C
      ENTRY NELOG(IPOCO,XDIST,LOG1,LOG2)
      LOG1=.FALSE.
      LOG2=.FALSE.
      DO 100 I=1,250
      J=I
 100  IF (INT(XALT(I,1)).EQ.IPOCO) GO TO 200
C    TO CHECK THAT ALL INFORMATION IS ON FILE PSOF.DD
      WRITE(32,120)IPOCO
 120  FORMAT(4X,'NO ALTERNATIVE FOUND FOR POSTCODE ',I5,' !!')
      RETURN ! ALTERNATIVE NOT FOUND
 200  IF (XALT(J,2).EQ.-1.0) GO TO 400
      IF (XDIST.LE.XALT(J,2)) LOG1=.TRUE.
      IF (XDIST.LE.XALT(J,3)) LOG2=.TRUE.
 400  RETURN
      END
C
C     MPLOT
C      WITH ENTRY PLOT2
C
      SUBROUTINE MPLOT
      COMMON/COD0/FNAME(7),IZOP(200),INOV(200),IZNO(200),COUNT(1),
     . IFORN,ASTAY,LOG1(200),LOG2(200),CFONO
      COMMON/COD5/JR,JS,JORI,XFCT(200),YCOS(200)
      CALL PAPUR (1,'PSOF1.GRD')
      RETURN
C
C    ENTRY PLOT2
C
      ENTRY PLOT2(INU)
      XMIN=0.10
```

```
42100          XMAX=0.40
42200          YMIN=0.10
42300          YMAX=0.40
42400          IPLO=INU-INT(INU/4.0)*4.0
42500          IF (IPLO.EQ.1) YFAC=0.5
42600          IF (IPLO.EQ.1) XFAC=0.0
42700          IF (IPLO.EQ.2) XFAC=0.0
42800          IF (IPLO.EQ.2) YFAC=0.0
42900          IF (IPLO.EQ.3) YFAC=0.5
43000          IF (IPLO.EQ.3) XFAC=0.5
43100          IF (IPLO.EQ.0) XFAC=0.5
43200          IF (IPLO.EQ.0) YFAC=0.0
43300          CALL PSPACE(XMIN+XFAC,XMAX+XFAC,YMIN+YFAC,YMAX+YFAC)
43400          CALL MAXMI(XPCT,YCOS,JS,XMIN2,XMAX2,YMIN2,YMAX2)
43500    C IF ONLY POINTS FOR WHICH DEMAND > 0 WANTS PLOTTED -- CHANGE TO JORI
43600    C  IN LINE 43300 , 43900 AND 46600
43700          CALL MAP(XMIN2,XMAX2*1.1,YMIN2,YMAX2*1.1)
43800          CALL CTRMAG(15)
43900          CALL PTPLOT(XPCT,YCOS,1,JS,45)
44000    C     DO 200 I=1,JS
44100    C200  CALL POINT(XPCT(I),YCOS(I))
44200           CALL BORDER
44300          J=0
44400          CALL CTRMAG(8)
44500           IF (XMAX2.GT.1.0) GO TO 250
44600       5  J=J+1 ! FOR CHECK ON X-AXIS
44700          IF ((10.0**J)*XMAX2.LT.1.0) GO TO 5
44800          XMAX3=XMAX2*(10.0**J)
44900          XINT3=FLOAT(INT(XMAX3+1.0))/10.0
45000          CALL MAP(XMIN2,XMAX3*1.1,YMIN2,YMAX2*1.1)
45100          CALL AXESSI(XINT3,0)
45200          GO TO 275
45300      250 CALL AXESS
45400      275 CALL CTRMAG(-1)
45500          CALL PSPACE(0.0+XFAC,0.5+XFAC,0.0+YFAC,0.5+YFAC)
45600          CALL MAP(0.0,100.0,0.0,100.0)
45700          CALL CTRMAG(15)
45800          CALL PLOTCS(15.0,90.0,FNAME,35)
45900    C     CALL CTRMAG(-1)
46000          CALL CTRMAG(10)
46100          CALL PLOTCS(45.0,10.0,'VISIT RATE',10)
46200          CALL POSITN(22.0,83.0)
46300          CALL TYPECS(COUNT,5)
46400          CALL SPACE(1)
46500          CALL TYPENI(CFONO)
46600          CALL SPACE(5)
46700           CALL TYPECS('ZONES :',7)
46800           CALL SPACE(1)
46900           CALL TYPENI(JS)
47000           IF (J.EQ.0) GO TO 300
47100           CALL POSITN(75.0,10.0)
47200           CALL TYPENI(10)
47300           CALL SUFFIX
47400           CALL TYPENI(-J)
47500           CALL NORMAL
47600      300  CALL CTRORI(1,0)
47700          CALL PLOTCS(6.0,40.0,'TRAVEL COST (KM)',16)
47800          CALL CTRORI(0,0)
47900    C     CALL CTRMAG(-1)
48000          IF (IPLO.EQ.0) CALL FRAME
```

```
48100           RETURN
48200     C
48300     C  ENTRY PLOTND
48400     C
48500           ENTRY PLOTND
48600           CALL GREND
48700           RETURN
48800           END
48900     C
49000     C       MAXMI
49100     C
49200           SUBROUTINE MAXMI(XARR,YARR,INOB,XMIN,XMAX,YMIN,YMAX)
49300           DIMENSION XARR(INOB),YARR(INOB)
49400           XMIN=0.0
49500           YMIN=0.0
49600           XMAX=0.0
49700           YMAX=0.0
49800           DO 100 I=1,INOB
49900           XMAX=AMAX1(XMAX,XARR(I))
50000           YMAX=AMAX1(YMAX,YARR(I))
50100           XMIN=AMIN1(XMIN,XARR(I))
50200           YMIN=AMIN1(YMIN,YARR(I))
50300      100  CONTINUE
50400           RETURN
50500           END
```

Appendix A5.3

Computer Programme PFAFO2.FOR and PFAFO3.FOR

```
00100    COMMENT PROGRAM PFAF02
00200    C       CREATES THE FOR30.DAT FILE FOR GENSTAT FOR A GIVEN FOREST
00300    C       AND SELECTS ON VARIABLES ALG1 AND ALG2
00400    C       READS FROM PFAF01.RES
00500    C       OUTPUT ON CHANNEL 27
00600    C
00700            PROGRAM PFAF02
00800            DIMENSION IX(200),IY(200),IZOP(200),XPCT(200),YCOS(200),ALG1(200)
00900           . ALG2(200),FNAM2(7),FNAME(7),COUNT(1)
01000            OPEN(UNIT=21,FILE='PFAF01.RES',ACCESS='SEQIN')
01100            ITTX=5;ITTY=27
01200          3 WRITE(ITTX,5)
01300          5 FORMAT(//,4X,'NAME OF FOREST ? ',$)
01400            REWIND 21
01500            READ(ITTX,7)(FNAM2(J),J=1,7)
01600          7 FORMAT(7A5)
01700            WRITE(ITTY,6)(FNAM2(J),J=1,7)
01800          6 FORMAT(//,4X,'NAME OF FOREST : ',7A5,/)
01900         45 READ(21,58,ERR=45,END=6666)(FNAME(I),I=1,7)
02000         58 FORMAT(15X,7A5)
02100            DO 55 I=1,2
02200         55 IF (FNAM2(I).NE.FNAME(I)) GO TO 45
02300            BACKSPACE 21
02400            READ(21,60)COUNT(1),CFONO,(FNAME(I),I=1,7),IFORN,ASTAT,JR,ITOT
02500         60 FORMAT(4X,A5,I3,3X,7A5,/,4X,I4,4X,F9.5,3X,I3,2X,I8,/)
02600            WRITE(ITTY,60)COUNT(1),CFONO,(FNAME(I),I=1,7),IFORN,ASTAT,JR,ITOT
02700            WRITE(ITTX,62)
02800         62 FORMAT(4X,'EXCLUDE OBS IF LOG1 = ',$)
02900            READ(ITTX,63)ALO1
03000         63 FORMAT(A1)
03100            WRITE(ITTX,64)
03200         64 FORMAT(4X,'EXCLUDE OBS IF LOG2 = ',$)
03300            READ(ITTX,63)ALO2
03400            WRITE(ITTY,66)ALO1,ALO2
03500         66 FORMAT(4X,'EXCLUDE OBS IF LOG1 = ',A1,/
03600           .       4X,'EXCLUDE OBS IF LOG2 = ',A2,/)
03700            JCOU=0
03800            DO 50 I=1,JR
03900            READ(21,70)IX(I),IY(I),IZOP(I),XPCT(I),YCOS(I),ALG1(I),ALG2(I)
04000         70 FORMAT(4X,I5,2X,I6,2X,I8,2X,F15.8,2X,F15.8,2(5X,A1))
04100            WRITE(ITTY,70)IX(I),IY(I),IZOP(I),XPCT(I),YCOS(I),ALG1(I),ALG2(I)
04200            IF (ALG1(I).EQ.ALO1) GO TO 50
04300            IF (ALG2(I).EQ.ALO2) GO TO 50
04400            JCOU=JCOU+1
04500            IX(JCOU)=IX(I)
04600            IY(JCOU)=IY(I)
04700            IZOP(JCOU)=IZOP(I)
04800            XPCT(JCOU)=XPCT(I)
04900            YCOS(JCOU)=YCOS(I)
05000            ALG1(JCOU)=ALG1(I)
05100            ALG2(JCOU)=ALG2(I)
05200         50 CONTINUE
05300            WRITE(30,300)(FNAME(I),I=1,7),ALO1,ALO2,JCOU
05400        300 FORMAT(2H'',7A5,2H'',/,2(2H'',A5,2H'',/),I3)
05500            DO 200 I=1,JCOU
05600        200 WRITE(30,450)YCOS(I),XPCT(I),IZOP(I),ALG1(I),ALG2(I) ! FOR GENST
05700        450 FORMAT(4X,2(F15.8,2X),I8,2X,2(A1,3X))
05800            WRITE(30,460)
05900        460 FORMAT(5H'EOD')
06000            WRITE(ITTY,5556)JCOU
```

```
06100      5556 FORMAT(//,4X,I3,X,'OBSERVATIONS WRITTEN TO FOR30.DAT ',/)
06200           STOP
06300      6666 WRITE(ITTY,6667)
06400      6667 FORMAT(4X,'FOREST NOT FOUND')
06500           GO TO 3
06600           STOP
06700           END
```

```
00100    COMMENT PROGRAM PFAF03
00200    C      DRAWS THE CURVES FROM PFAF01 AND GENSTAT RESULTS
00300    C      READS FROM PFAF01.RES
00400    C      INTEGRATES CONSUMERS SURPLUS AND NO OF VISITORGROUP PREDICTED
00500    C        AT ZERO PRICE
00600    C        DRAWS THE NET DEMAND CURVE
00700    C      IT IS AN INTERACTIVE VERSION; ASK FOR NAME OF A PARTICULAR
00800    C        FOREST, WHICH IS THEN FOUND ;
00900    C        READS THE PARAMETERS ESTIMATED IN GENSTAT FROM FOR31.DAT
01000    C        AND CALCULATES THE RESULTS
01100    C        OUTPUT ON CHANNEL 28
01200    C
01300          PROGRAM PFAF03
01400          DIMENSION IX(200),IY(200),FNAM2(7),ALG1(200),ALG2(200),XCS(200),
01500         . XNO(200)
01600          COMMON/COD0/FNAME(7),IZOP(200),INOV(200),IZNO(200),COUNT(1),
01700         . IFORN,ASTAY,LOG1(200),LOG2(200),CFONO
01800          COMMON/COD5/JR,JS,XFCT(200),YCOS(200)
01900          COMMON/COD6/XA1,XB1,XA2,XB2,ALO1,ALO2
02000          OPEN(UNIT=21,FILE='PFAF01.RES',ACCESS='SEQIN')
02100          OPEN(UNIT=22,FILE='FOR31.DAT')
02200          ITTX=5;ITTY=28
02300       3  WRITE(ITTX,5)
02400       5  FORMAT(4X,'NAME OF FOREST ? ',$)
02500          READ(ITTX,7)(FNAM2(J),J=1,7)
02600       7  FORMAT(7A5)
02700          WRITE(ITTY,6)(FNAM2(J),J=1,7)
02800       6  FORMAT(/,4X,'NAME OF FOREST : ',7A5,/)
02900          REWIND 21
03000      45  READ(21,58,ERR=45,END=6666)(FNAME(I),I=1,7)
03100      58  FORMAT(15X,7A5)
03200          DO 55 I=1,2
03300      55  IF (FNAM2(I).NE.FNAME(I)) GO TO 45.
03400          BACKSPACE 21
03500          READ(21,60)COUNT(1),CFONO,(FNAME(I),I=1,7),IFORN,ASTAT,JR,ITOT
03600      60  FORMAT(4X,A5,I3,3X,7A5,/,4X,I4,4X,F9.5,3X,I3,2X,I8,/)
03700          WRITE(ITTY,60)COUNT(1),CFONO,(FNAME(I),I=1,7),IFORN,ASTAT,JR,ITO
03800          WRITE(ITTX,72)
03900      72  FORMAT(/,4X,'PARAMETERS FOR EXPFIT TO BE READ FROM FILE (YES OR
04000         .) ? ',$)
04100          READ(ITTX,74)ANSW1
04200      74  FORMAT(A3)
04300          IF (ANSW1.EQ.'YES') GOTO 79
04400          WRITE(ITTX,75)
04500      75  FORMAT(//,4X,'PARAMETERS FROM 1st STAGE EXPFIT:',//,
04600         . 8X,'A1 = ',$)
04700          READ(ITTX,*)XA1
04800          WRITE(ITTX,76)
04900      76  FORMAT(8X,'B1 = ',$)
05000          READ(ITTX,*)XB1
05100          WRITE(ITTX,77)
05200      77  FORMAT(//,4X,'PARAMETERS FROM 2nd STAGE EXPFIT:',//,
05300         . 8X,'A2 = ',$)
05400          READ(ITTX,*)XA2
05500          WRITE(ITTX,78)
05600      78  FORMAT(8X,'B2 = ',$)
05700          READ(ITTX,*)XB2
05800          GO TO 81
05900      79  READ(22,90)XA1,XB1,XA2,XB2
06000      90  FORMAT(/,X,2(13X,F12.8),/,/,X,2(13X,F12.8))
```

```
06100       81  WRITE(ITTY,80)XA1,XB1
06200       80  FORMAT(//,4X,'PARAMETERS FROM 1st STAGE EXPFIT:',//,
06300          . 8X,'A1 = ',F12.8,/,8X,'A2 = ',F12.8,/)
06400           WRITE(ITTY,85)XA2,XB2
06500       85  FORMAT(//,4X,'PARAMETERS FROM 2nd STAGE EXPFIT:',//,
06600          . 8X,'A2 = ',F12.8,/,8X,'B2 = ',F12.8,/)
06700           WRITE(ITTX,62)
06800       62  FORMAT(4X,'EXCLUDE OBS IF LOG1 = ',$)
06900           READ(ITTX,63)ALO1
07000       63  FORMAT(A1)
07100           WRITE(ITTX,64)
07200       64  FORMAT(4X,'EXCLUDE OBS IF LOG2 = ',$)
07300           READ(ITTX,63)ALO2
07400           WRITE(ITTY,66)ALO1,ALO2
07500       66  FORMAT(/,4X,'EXCLUDE IF LOG1 = ',A1,/
07600                  ,4X,'EXCLUDE IF LOG2 = ',A1,/)
07700           JCOU=0
07800           ITOT=0
07900           DO 50 I=1,JR
08000           READ(21,70)IX(I),IY(I),IZOP(I),XPCT(I),YCOS(I),ALG1(I),ALG2(I)
08100       70  FORMAT(4X,I5,2X,I6,2X,I8,2X,F15.8,2X,F15.8,2(5X,A1))
08200           IF (ALG1(I).EQ.ALO1) GO TO 50
08300           IF (ALG2(I).EQ.ALO2) GO TO 50
08400           JCOU=JCOU+1
08500           IX(JCOU)=IX(I)
08600           IY(JCOU)=IY(I)
08700           IZOP(JCOU)=IZOP(I)
08800           XPCT(JCOU)=XPCT(I)
08900           YCOS(JCOU)=YCOS(I)
09000           ALG1(JCOU)=ALG1(I)
09100           ALG2(JCOU)=ALG2(I)
09200           ITOT=ITOT+IY(JCOU)
09300        50 CONTINUE
09400           WRITE(ITTY,250)JCOU,JR
09500       250 FORMAT(/,4X,I3,' OBSERVATIONS OUT OF ',I3,' SELECTED',/)
09600           JR=JCOU
09700           CALL MPLOT
09800           CALL PLOT2(1)
09900           CALL PLOT3(2)
10000           CALL PLOND
10100       C     CALCULATES CONSUMERS SURPLUS AND NO OF PREDICTED VISITORGROUPS
10200           XTOT=0.0 ; XNTO=0.0
10300           WRITE(ITTY,490)
10400       490 FORMAT(//,10X,'ZONE',3X,'NO OBS.',3X,'NO PRED.',4X,'CONS. SURPL.'
10500          . /)
10600           DO 500 I=1,JR
10700           XCS(I)=(-XA2/XB2*EXP(XB2*YCOS(I)))*FLOAT(IZOP(I))
10800           XNO(I)=(XA2*EXP(XB2*YCOS(I)))*FLOAT(IZOP(I))
10900           XNTO=XNTO+XNO(I)    ! SUMS NO OF VISITORGROUPS
11000           XTOT=XTOT+XCS(I)    ! SUMS CONSUMERS SURPLUS
11100           ITMP=INT(XNO(I))
11200       500 WRITE(ITTY,505)IX(I),IY(I),ITMP,XCS(I)
11300       505 FORMAT(4X,3(4X,I6),4X,F12.0)
11400           INNO=INT(XNTO+0.5)
11500           WRITE(ITTY,100)XTOT,INNO,ITOT
11600       100 FORMAT(//,4X,'CONSUMERS SURPLUS EQUALS  ',F15.0,/,
11700                  /,4X,'PREDICTED NO. OF VISITORGROUPS      ',I14,/,
11800          .         4X,'ACTUAL OBSERVED NO. OF VISITORGROUPS',I14,//)
11900           WRITE(ITTX,5558)
12000       5558 FORMAT(//,4X,'NEW FOREST (YES OR NO) ? ',$)
```

```
12100           READ(ITTX,74)ANSW2
12200           IF (ANSW2,EQ,'YES') GO TO 3
12300           GO TO 666
12400      6666 WRITE(ITTY,6667)
12500      6667 FORMAT(4X,'FOREST NOT FOUND ',/)
12600           GO TO 3
12700       666 STOP
12800           END
12900     C
13000     C     MYFN1
13100     C
13200           REAL FUNCTION MYFN1(X)
13300           REAL X
13400           COMMON/COD6/XA1,XB1,XA2,XB2
13500           MYFN1=XA1*EXP(XB1*X)
13600           RETURN
13700           END
13800     C
13900     C     MYFN2
14000     C
14100           REAL FUNCTION MYFN2(X)
14200           MYFN2=X
14300           RETURN
14400           END
14500     C
14600     C     MYFN3
14700     C
14800           REAL FUNCTION MYFN3(X)
14900           COMMON/COD6/XA1,XB1,XA2,XB2
15000           REAL X
15100           MYFN3=XA2*EXP(XB2*X)
15200           RETURN
15300           END
15400     C
15500     C     MPLOT
15600     C      WITH ENTRY PLOT2 AND PLOT3
15700     C
15800           SUBROUTINE MPLOT
15900           DIMENSION XVIS(500),YADM(500)
16000           COMMON/COD0/FNAME(7),IZOP(200),INOV(200),IZNO(200),COUNT(1),
16100          . IFORN,ASTAY,LOG1(200),LOG2(200),CFONO
16200           COMMON/COD5/JR,JS,XPCT(200),YCOS(200)
16300           COMMON/COD6/XA1,XB1,XA2,XB2,ALO1;ALO2
16400           DOUBLE PRECISION NAM
16500           EXTERNAL MYFN1,MYFN2,MYFN3
16600           ITTX=5;ITTY=28
16700           WRITE(ITTX,10)
16800        10 FORMAT(/,4X,'NAME FOR GRIDFILE ? ',$)
16900           READ(ITTX,15)NAM
17000        15 FORMAT(A10)
17100           WRITE(ITTY,16)NAM
17200        16 FORMAT(/,4X,'GRIDFILE = ',A10,/)
17300           CALL PAPUR (1,NAM)
17400           RETURN
17500     C  ENTRY PLOT2
17600           ENTRY PLOT2(INU)
17700           XMIN=0.10
17800           XMAX=0.40
17900           YMIN=0.10
18000           YMAX=0.40
```

```
18100           IPLO=INU-INT(INU/4.0)*4.0
18200           IF (IPLO.EQ.1) YFAC=0.5
18300           IF (IPLO.EQ.1) XFAC=0.0
18400           IF (IPLO.EQ.2) XFAC=0.0
18500           IF (IPLO.EQ.2) YFAC=0.0
18600           IF (IPLO.EQ.3) YFAC=0.5
18700           IF (IPLO.EQ.3) XFAC=0.5
18800           IF (IPLO.EQ.0) XFAC=0.5
18900           IF (IPLO.EQ.0) YFAC=0.0
19000           CALL PSPACE(XMIN+XFAC,XMAX+XFAC,YMIN+YFAC,YMAX+YFAC)
19100           CALL MAXMI(XPCT,YCOS,JR,XMIN2,XMAX2,YMIN2,YMAX2)
19200           CALL MAP(XMIN2,XMAX2*1.1,YMIN2,YMAX2*1.1)
19300           CALL CTRMAG(15)
19400           CALL PTPLOT(XPCT,YCOS,1,JR,45)
19500           YINT3=((YMAX2-YMIN2)/300.0)
19600           CALL BROKEN(8,4,4,4)
19700           CALL LINEF(MYFN1,MYFN2,YMIN2,YINT3,YMAX2)
19800           CALL FULL
19900           CALL LINEF(MYFN3,MYFN2,YMIN2,YINT3,YMAX2)
20000            CALL BORDER
20100           J=0
20200           CALL CTRMAG(8)
20300            IF (XMAX2.GT.1.0) GO TO 250
20400       5   J=J+1 ! FOR CHECK ON X-AXIS
20500           IF ((10.0**J)*XMAX2.LT.1.0) GO TO 5
20600           XMAX3=XMAX2*(10.0**J)
20700           XINT3=FLOAT(INT(XMAX3+1.0))/10.0
20800           CALL MAP(XMIN2,XMAX3*1.1,YMIN2,YMAX2*1.1)
20900           CALL AXESSI(XINT3,0)
21000           GO TO 275
21100       250 CALL AXESS
21200       275 CALL CTRMAG(-1)
21300           CALL PSPACE(0.0+XFAC,0.5+XFAC,0.0+YFAC,0.5+YFAC)
21400           CALL MAP(0.0,100.0,0.0,100.0)
21500           CALL CTRMAG(15)
21600           CALL PLOTCS(15.0,90.0,FNAME,35)
21700           CALL CTRMAG(10)
21800           CALL PLOTCS(45.0,10.0,'VISIT RATE',10)
21900           CALL POSITN(21.0,83.0)
22000           CALL TYPECS(COUNT,5)
22100           CALL SPACE(1)
22200           CALL TYPENI(CFOND)
22300           CALL SPACE(3)
22400            CALL TYPECS('ZONES :',7)
22500            CALL TYPENI(JR)
22600            CALL SPACE(3)
22700            ALO11=ALO1
22800            ALO22=ALO2
22900            IF (ALO11.EQ.'T') GO TO 280
23000            IF (ALO11.EQ.'F') ALO11='T'
23100            GO TO 282
23200       280 IF (ALO11.EQ.'T') ALO11='F'
23300       282 IF (ALO22.EQ.'T') GO TO 284
23400            IF (ALO22.EQ.'F') ALO22='T'
23500            GO TO 286
23600       284 IF (ALO22.EQ.'T') ALO22='F'
23700       286 CALL TYPECS('L1:',3)
23800            CALL TYPECS(ALO11,1)
23900            CALL SPACE(1)
24000            CALL TYPECS('L2:',3)
```

```
24100           CALL TYPECS(ALO22,1)
24200           IF (J.EQ.0) GO TO 288
24300           CALL POSITN(75.0,10.0)
24400           CALL TYPENI(10)
24500           CALL SUFFIX
24600           CALL TYPENI(-J)
24700           CALL NORMAL
24800       288 CALL CTRMAG(8)
24900           CALL POSITN(48.0,70.0)
25000           CALL BROKEN(8,4,4,4)
25100           CALL JOIN(52.0,70.0)
25200           CALL FULL
25300           CALL SPACE(1)
25400           CALL TYPECS('Y=',2)
25500           CALL TYPENF(XA1,3)
25600           CALL TYPECS('*',1)
25700           CALL CTRSET(6)
25800           CALL CTRMAG(10)
25900           CALL TYPECS('E',1)
26000           CALL CTRSET(0)
26100           CALL SUFFIX
26200           CALL TYPENF(XB1,3)
26300           CALL TYPECS('*',1)
26400           CALL TYPECS('X',1)
26500           CALL NORMAL
26600           CALL CTRMAG(8)
26700           CALL POSITN(48.0,66.0)
26800           CALL JOIN(52.0,66.0)
26900           CALL SPACE(1)
27000           CALL TYPECS('Y=',2)
27100           CALL TYPENF(XA2,3)
27200           CALL TYPECS('*',1)
27300           CALL CTRSET(6)
27400           CALL CTRMAG(10)
27500           CALL TYPECS('E',1)
27600           CALL CTRSET(0)
27700           CALL SUFFIX
27800           CALL TYPENF(XB2,3)
27900           CALL TYPECS('*',1)
28000           CALL TYPECS('X',1)
28100           CALL NORMAL
28200           CALL CTRMAG(8)
28300           CALL POSITN(80.0,16.5)
28400           CALL TYPECS('(Y)',3)
28500           CALL POSITN(15.5,81.0)
28600           CALL TYPECS('(X)',3)
28700           CALL CTRMAG(10)
28800           CALL CTRSET(0)
28900          CALL CTRORI(1,0)
29000           CALL PLOTCS(6.0,40.0,'TRAVEL COST (KM)',16)
29100           CALL CTRORI(0,0)
29200           IF (IPLO.EQ.0) CALL FRAME
29300           RETURN
29400     C   ENTRY PLOT3
29500           ENTRY PLOT3(INU)
29600           XMIN=0.10
29700           XMAX=0.40
29800           YMIN=0.10
29900           YMAX=0.40
30000           IPLO=INU-INT(INU/4.0)*4.0
```

```
30100           IF (IPLO.EQ.1) YFAC=0.5
30200           IF (IPLO.EQ.1) XFAC=0.0
30300           IF (IPLO.EQ.2) XFAC=0.0
30400           IF (IPLO.EQ.2) YFAC=0.0
30500           IF (IPLO.EQ.3) YFAC=0.5
30600           IF (IPLO.EQ.3) XFAC=0.5
30700           IF (IPLO.EQ.0) XFAC=0.5
30800           IF (IPLO.EQ.0) YFAC=0.0
30900           CALL PSPACE(XMIN+XFAC,XMAX+XFAC,YMIN+YFAC,YMAX+YFAC)
31000           CALL MAXMI(XPCT,YCOS,JR,XMIN2,XMAX2,YMIN2,YMAX2)
31100           DO 100 J=1,500
31200           XVIS(J)=0.0
31300     100   YADM(J)=0.0
31400           KMAX=0
31500           YMAX3=YMAX2
31600           IF (YMAX3.LT.100.0) YMAX3=100.0
31700           XSTEP=(YMAX3-YMIN2)/400.0
31800           IF (XSTEP.LT.0.1) XSTEP=0.1
31900           DO 300 I=1,JR
32000           K=0
32100           DO 200 XJ=YCOS(I),YMAX3,XSTEP
32200           K=K+1
32300     200   XVIS(K)=XVIS(K)+(XA2*EXP(XB2*XJ)*FLOAT(IZOP(I)))
32400     300   IF (KMAX.LT.K) KMAX=K
32500           DO 350 I=1,KMAX
32600     350   YADM(I)=FLOAT((I-1))*XSTEP
32700           CALL MAXMI(XVIS,YADM,KMAX,XMIN4,XMAX4,YMIN4,YMAX4)
32800           CALL MAP(0.0,XMAX4*1.1,0.0,YMAX4*1.1)
32900           CALL CTRMAG(15)
33000            CALL BORDER
33100            CALL POSITN(XVIS(1),YADM(1))
33200            DO 600 I=2,KMAX
33300     600    CALL JOIN(XVIS(I),YADM(I))
33400           J=0
33500           CALL CTRMAG(8)
33600            IF (XMAX4.LT.10.0) GO TO 1250
33700     1005  J=J+1 ! FOR CHECK ON X-AXIS
33800           IF (XMAX4/(10.0**J).GT.10.0) GO TO 1005
33900           XMAX3=XMAX4/(10.0**J)
34000           XINT3=FLOAT(INT(XMAX3+1.0))/10.0
34100           CALL MAP(0.0,XMAX3*1.1,0.0,YMAX4*1.1)
34200           CALL AXESSI(XINT3,0)
34300           GO TO 1275
34400     1250   CALL AXESS
34500     1275   CALL CTRMAG(-1)
34600           CALL PSPACE(0.0+XFAC,0.5+XFAC,0.0+YFAC,0.5+YFAC)
34700           CALL MAP(0.0,100.0,0.0,100.0)
34800           CALL CTRMAG(15)
34900           CALL PLOTCS(15.0,90.0,FNAME,35)
35000           CALL CTRMAG(10)
35100           CALL PLOTCS(35.0,10.0,'NO OF VISITORGROUPS',19)
35200           CALL POSITN(21.0,83.0)
35300           CALL TYPECS(COUNT,5)
35400           CALL SPACE(1)
35500           CALL TYPENI(CFONO)
35600           CALL SPACE(3)
35700            CALL TYPECS('ZONES :',7)
35800            CALL TYPENI(JR)
35900            CALL SPACE(3)
36000            ALO11=ALO1
```

```
36100            ALO22=ALO2
36200            IF (ALO11.EQ.'T') GO TO 1280
36300            IF (ALO11.EQ.'F') ALO11='T'
36400            GO TO 1282
36500       1280 IF (ALO11.EQ.'T') ALO11='F'
36600       1282 IF (ALO22.EQ.'T') GO TO 1284
36700            IF (ALO22.EQ.'F') ALO22='T'
36800            GO TO 1286
36900       1284 IF (ALO22.EQ.'T') ALO22='F'
37000       1286 CALL TYPECS('L1:',3)
37100            CALL TYPECS(ALO11,1)
37200            CALL SPACE(1)
37300            CALL TYPECS('L2:',3)
37400            CALL TYPECS(ALO22,1)
37500            IF (J.EQ.0) GO TO 310
37600            CALL POSITN(75.0,10.0)
37700            CALL TYPENI(10)
37800            CALL SUFFIX
37900            CALL TYPENI(J)
38000            CALL NORMAL
38100        310 CALL CTRMAG(10)
38200            CALL CTRSET(0)
38300           CALL CTRORI(1,0)
38400           CALL PLOTCS(6.0,38.0,'ADMISSION FEE (KM)',18)
38500           CALL CTRORI(0,0)
38600           IF (IPLO.EQ.0) CALL FRAME
38700           RETURN
38800       C ENTRY PLOTND
38900            ENTRY PLOND
39000            CALL GREND
39100            RETURN
39200            END
39300       C
39400       C    MAXMI
39500       C
39600            SUBROUTINE MAXMI(XARR,YARR,INOB,XMIN,XMAX,YMIN,YMAX)
39700            DIMENSION XARR(INOB),YARR(INOB)
39800            XMIN=0.0
39900            YMIN=0.0
40000            XMAX=0.0
40100            YMAX=0.0
40200            DO 100 I=1,INOB
40300            XMAX=AMAX1(XMAX,XARR(I))
40400            YMAX=AMAX1(YMAX,YARR(I))
40500            XMIN=AMIN1(XMIN,XARR(I))
40600            YMIN=AMIN1(YMIN,YARR(I))
40700        100 CONTINUE
40800            RETURN
40900            END
```

Appendix A5.4

Example of Output

START User F349 JBCH [146,24155] Job HARE11 Seq. 5319 Date 15-Dec-81 12:52:18

```
FFFFF   333    4   4     999                J  BBBB     CCCC  H    H
F      3   3   4   4    9   9               J  B   B   C      H    H
F          3   4   4    9   9               J  B   B   C      H    H
FFFF       3   44444    9999               J  BBBB    C      HHHHH
F          3       4        9          J   J  B   B   C      H    H
F      3   3       4        9          J   J  B   B   C      H    H
F       333        4    999             JJJ   BBBB     CCCC  H    H
```

START User F349 JBCH [146,24155] Job HARE11 Seq. 5319 Date 15-Dec-81 12:52:18

START User F349 JBCH [146,24155] Job HARE11 Seq. 5319 Date 15-Dec-81 12:52:18

START User F349 JBCH [146,24155] Job HARE11 Seq. 5319 Date 15-Dec-81 12:52:18

```
N    N   OOO   TTTTT  EEEEE                H    H   AAA   RRRR   EEEEE   SSSS  K   K
NN   N  O   O    T    E      ::            H    H  A   A  R   R  E      S      K  K
N N  N  O   O    T    E      ::            H    H  A   A  R   R  E      S      K K
N  N N  O   O    T    EEEE                 HHHHHH  A   A  RRRR   EEEE    SSS   KKK
N   NN  O   O    T    E      ::            H    H  AAAAA  R R    E          S  K K
N    N  O   O    T    E      ::            H    H  A   A  R  R   E          S  K  K
N    N   OOO     T    EEEEE                H    H  A   A  R   R  EEEEE  SSSS   K   K
```

START User F349 JBCH [146,24155] Job HARE11 Seq. 5319 Date 15-Dec-81 12:52:18

START User F349 JBCH [146,24155] Job HARE11 Seq. 5319 Date 15-Dec-81 12:52:18

```
!
"0000000000000000000000000000000000000000000000000000000000000000000000000000
#0000000011111111112222222223333333334444444444555555555566666666667777777788
$1234567890123456789012345678901234567890123456789012345678901234567890123456
```

NAME OF FOREST : HARESKOVENE

01KBH 10 HARESKOVENE OG JONSTRUP VANG
.8 1.32804 70 59104

EXCLUDE OBS IF LOG1 = A
EXCLUDE OBS IF LOG2 = A

1000	296	22424	0.01320014	36.80000000	F	T
1500	500	39361	0.01270293	34.00000000	F	T
2000	945	29628	0.03189550	31.20000000	T	T
2100	1508	47598	0.03168200	37.60000000	F	T
2200	392	43698	0.00897066	24.40000000	T	T
2300	152	52995	0.00286820	39.60000000	F	T
2400	657	26948	0.02438029	32.00000000	F	T
2450	277	10410	0.02660903	37.40000000	T	T
2500	71	27546	0.00257751	34.80000000	T	T
2600	1366	13498	0.10120018	31.80000000	T	T
2610	221	17963	0.01230307	27.60000000	T	T
2620	3168	12148	0.26078367	31.60000000	T	T
2630	829	11209	0.07395843	34.40000000	F	T
2635	120	6794	0.01766264	40.00000000	T	T
2640	184	4472	0.04114490	49.60000000	T	T
2650	807	23600	0.03419492	37.20000000	T	T
2660	63	6580	0.00957447	46.00000000	T	T
2670	377	8385	0.04496124	51.00000000	T	T
2700	1516	22949	0.06605952	22.40000000	T	T
2720	389	16789	0.02316993	26.00000000	T	T
2730	4823	13905	0.34685365	17.40000000	T	T
2740	4603	5309	0.86701827	16.40000000	T	T
2750	16966	12223	1.38803890	10.00000000	T	T
2760	3252	4294	0.75733582	11.60000000	T	T
2770	290	16413	0.01766892	48.00000000	F	T
2800	127	18161	0.00699301	15.80000000	T	T
2820	298	10000	0.02980000	22.80000000	F	T
2860	1037	17108	0.06061492	17.20000000	T	T
2880	1156	7488	0.15438034	9.40000000	T	T
2900	1207	10488	0.11508391	28.20000000	F	T
3400	88	11545	0.00762235	42.00000000	F	F
3500	10369	6368	1.62829770	5.40000000	T	T
3520	696	5820	0.11958763	14.60000000	F	T
3650	248	3759	0.06597499	35.80000000	F	T
3660	50	2288	0.02185315	27.40000000	F	T
4621	56	1041	0.05379443	69.60000000	F	F
2680	0	3578	0.00000000	66.40000000	F	F
2690	0	2362	0.00000000	56.60000000	F	T
2791	0	5731	0.00000000	61.60000000	F	F
2830	0	8440	0.00000000	24.60000000	F	T
2840	0	6055	0.00000000	26.00000000	F	T
2850	0	3125	0.00000000	29.00000000	F	F
2920	0	10634	0.00000000	26.80000000	F	T
2930	0	2088	0.00000000	26.80000000	F	T
2942	0	804	0.00000000	34.00000000	F	T
2950	0	2697	0.00000000	36.40000000	F	F
2960	0	2171	0.00000000	49.40000000	F	F
2970	0	6419	0.00000000	43.80000000	F	F
2980	0	3549	0.00000000	48.00000000	F	F

2990	0	1567	0.00000000	57.80000000	F	F
3050	0	3382	0.00000000	61.80000000	F	F
3060	0	3869	0.00000000	64.40000000	F	F
3030	0	444	0.00000000	69.00000000	F	F
3320	0	1176	0.00000000	53.40000000	F	F
3330	0	370	0.00000000	44.40000000	F	F
3340	0	232	0.00000000	36.80000000	F	F
3400	0	11545	0.00000000	45.20000000	F	F
3450	0	5600	0.00000000	28.40000000	F	F
3460	0	7751	0.00000000	25.80000000	F	T
3471	0	45	0.00000000	25.20000000	F	F
3480	0	3937	0.00000000	63.20000000	F	F
3490	0	530	0.00000000	64.40000000	F	F
3530	0	35	0.00000000	20.00000000	F	T
3540	0	1407	0.00000000	30.20000000	F	F
3550	0	2442	0.00000000	42.80000000	F	F
3600	0	5544	0.00000000	57.80000000	F	F
3630	0	2306	0.00000000	69.60000000	F	F
3670	0	733	0.00000000	23.80000000	F	T
3310	0	631	0.00000000	68.40000000	F	F
4000	0	27051	0.00000000	63.80000000	F	F

70 OBSERVATIONS WRITTEN TO FOR30.DAT

GENSTAT V MARK 4.03
(C)1980 LAWES AGRICULTURAL TRUST (ROTHAMSTED EXPERIMENTAL STATION)

```
 1  'REFERENCE' EXPW01
 2  '' PROGRAM GENSTAT 00   1981-09-16 ''
 3  '' STEP 1  EXPFIT TO ESTIMATE VARIANS  ''
 4  'SCALAR' ILEN
 5  'HEAD' HEA2
 6  'HEAD' HL01
 7  'HEAD' HL02
 8  'INPUT' 2
 9  'READ/S' HEA2
10  'READ/S' HL01
11  'READ/S' HL02
12  'READ' ILEN
13  'INPUT' 1
14  'PRINT' HEA2
15  'PRINT' HL01,HL02
16  'PRINT' ILEN
17  'RUN'
```

HARESKOVENE OG JONSTRUP VANG

A

A

ILEN 7.0000E 1

```
18  'UNIT' MYUB $ ILEN
19  'VARIATE' COST $ ILEN
20  'VARIATE' VISI $ ILEN
```

```
'VARIATE' IZOP $ ILEN
'VARIATE' RES1 $ ILEN
'NAMES' VARL=T,F
'FACTOR' LOG1 $ VARL,ILEN
'FACTOR' LOG2 $ VARL,ILEN
'INPUT' 2
'READ/P' COST,VISI,IZOP,LOG1,LOG2
'INPUT' 1
'PRINT' HEA2
'PRINT/P' COST,VISI,IZOP,LOG1,LOG2
'MODEL' MYFUN $ MR=A*EXP(B*COST)
'SCALAR' A=5.5 : B=-0.070
'VARIATE' STEPL= 0.01 , 0.005
'OPTIMIZE / LIK= 2 , NPAR= 2 , PRINT=MPF ,ITER=50 '  MODEL= MYFUN
 ; PARAM=A,B;STEPS=STEPL;Y=VISI;Z=MR ; RES = RES1
'OUTPUT' 2
'PRINT/P' A,B $ 12.8 , 12.8
'OUTPUT' 1
'RUN'
```

COST	MNMIMAX	37.7514	5.4000	69.6000	70	VALUES
VISI	MNMIMAX	0.0926	0.0000	1.6283	70	VALUES
IZOP	MNMIMAX	10420.7858	35.0000	52995.0000	70	VALUES

HARESKOVENE OG JONSTRUP VANG

COST	VISI	IZOP	LOG1	LOG2
3.6800E 1	1.3200E -2	2.2424E 4	F	T
3.4000E 1	1.2703E -2	3.9361E 4	F	T
3.1200E 1	3.1896E -2	2.9628E 4	T	T
3.7600E 1	3.1682E -2	4.7598E 4	F	T
2.4400E 1	8.9707E -3	4.3698E 4	T	T
3.9600E 1	2.8682E -3	5.2995E 4	F	T
3.2000E 1	2.4380E -2	2.6948E 4	F	T
3.7400E 1	2.6609E -2	1.0410E 4	T	T
3.4800E 1	2.5775E -3	2.7546E 4	T	T
3.1800E 1	1.0120E -1	1.3498E 4	T	T
2.7600E 1	1.2303E -2	1.7963E 4	T	T
3.1600E 1	2.6078E -1	1.2148E 4	T	T
3.4400E 1	7.3958E -2	1.1209E 4	F	T
4.0000E 1	1.7663E -2	6.7940E 3	T	T
4.9600E 1	4.1145E -2	4.4720E 3	T	T
3.7200E 1	3.4195E -2	2.3600E 4	T	T
4.6000E 1	9.5745E -3	6.5800E 3	T	T
5.1000E 1	4.4961E -2	8.3850E 3	T	T
2.2400E 1	6.6060E -2	2.2949E 4	T	T
2.6000E 1	2.3170E -2	1.6789E 4	T	T
1.7400E 1	3.4685E -1	1.3905E 4	T	T
1.6400E 1	8.6702E -1	5.3090E 3	T	T
1.0000E 1	1.3380E 0	1.2223E 4	T	T
1.1600E 1	7.5734E -1	4.2940E 3	T	T
4.8000E 1	1.7669E -2	1.6413E 4	F	T
1.5800E 1	6.9930E -3	1.8161E 4	T	T
2.2800E 1	2.9300E -2	1.0000E 4	F	T
1.7200E 1	6.0615E -2	1.7105E 4	T	T
9.4000E 0	1.5438E -1	7.4880E 3	T	T
2.8200E 1	1.1508E -1	1.0488E 4	F	T
4.2000E 1	7.6224E -3	1.1545E 4	F	F

5.4000E	0	1.6283E	0	6.3680E	3	T		T
1.4600E	1	1.1959E	-1	5.8200E	3	F		T
3.5800E	1	6.5975E	-2	3.7590E	3	F		T
2.7400E	1	2.1853E	-2	2.2880E	3	F		T
6.9600E	1	5.3794E	-2	1.0410E	3	F		F
6.6400E	1	0.0000E	0	3.5780E	3	F		F
5.6600E	1	0.0000E	0	2.3620E	3	F		T
6.1600E	1	0.0000E	0	5.7310E	3	F		F
2.4600E	1	0.0000E	0	8.4400E	3	F		T
2.6000E	1	0.0000E	0	6.0550E	3	F		T
2.9000E	1	0.0000E	0	3.1250E	3	F		F
2.6800E	1	0.0000E	0	1.0634E	4	F		T
2.6800E	1	0.0000E	0	2.0880E	3	F		T
3.4000E	1	0.0000E	0	8.0400E	2	F		T
3.6400E	1	0.0000E	0	2.6970E	3	F		F
4.9400E	1	0.0000E	0	2.1710E	3	F		F
4.3800E	1	0.0000E	0	6.4190E	3	F		F
4.8000E	1	0.0000E	0	3.5490E	3	F		F
5.7800E	1	0.0000E	0	1.5670E	3	F		F
6.1800E	1	0.0000E	0	3.3820E	3	F		F
6.4400E	1	0.0000E	0	3.8690E	3	F		F
6.9000E	1	0.0000E	0	4.4400E	2	F		F
5.3400E	1	0.0000E	0	1.1760E	3	F		F
4.4400E	1	0.0000E	0	3.7000E	2	F		F
3.6800E	1	0.0000E	0	2.3200E	2	F		F
4.5200E	1	0.0000E	0	1.1545E	4	F		F
2.8400E	1	0.0000E	0	5.6000E	3	F		F
2.5800E	1	0.0000E	0	7.7510E	3	F		T
2.5200E	1	0.0000E	0	4.5000E	1	F		F
6.3200E	1	0.0000E	0	3.9370E	3	F		F
6.4400E	1	0.0000E	0	5.3000E	2	F		F
2.0000E	1	0.0000E	0	3.5000E	1	F		T
3.0200E	1	0.0000E	0	1.4070E	3	F		F
4.2800E	1	0.0000E	0	2.4420E	3	F		F
5.7800E	1	0.0000E	0	5.5440E	3	F		F
6.9600E	1	0.0000E	0	2.3060E	3	F		F
2.3800E	1	0.0000E	0	7.3300E	2	F		T
6.8400E	1	0.0000E	0	6.3100E	2	F		F
6.3800E	1	0.0000E	0	2.7051E	4	F		F

34.....;.....

*** CONVERGENCE MONITORING ***

ITER	FNO.	MOVE	FUNC VALUE	CURRENT PARAMETERS	
			STEPS	0.010000	0.005000
0	1	0	47.04281998	5.500000	-0.070000
			STEPS	0.082066	0.000609
1	15	2	2.59108275	1.735620	-0.083163
2	24	0	1.62790074	3.672650	-0.150375
3	33	0	1.58411105	3.941906	-0.164623
4	42	0	1.58256640	3.997415	-0.167810
			STEPS	0.010614	0.000294
5	51	1	1.58256376	3.999965	-0.167955
6	61	6	1.58256376	3.999686	-0.167947
			STEPS	0.065267	0.001812
1	70	0	1.58256376	3.999686	-0.167947

*** PARAMETER ESTIMATES ***

	ESTIMATE	S.E.	CORRELATIONS	
A	3.99969	0.65137	1.0000	
B	-0.16795	0.01808	-0.8997	1.0000

	DF	SS	MS
RESIDUAL	68	1.582564	0.023273

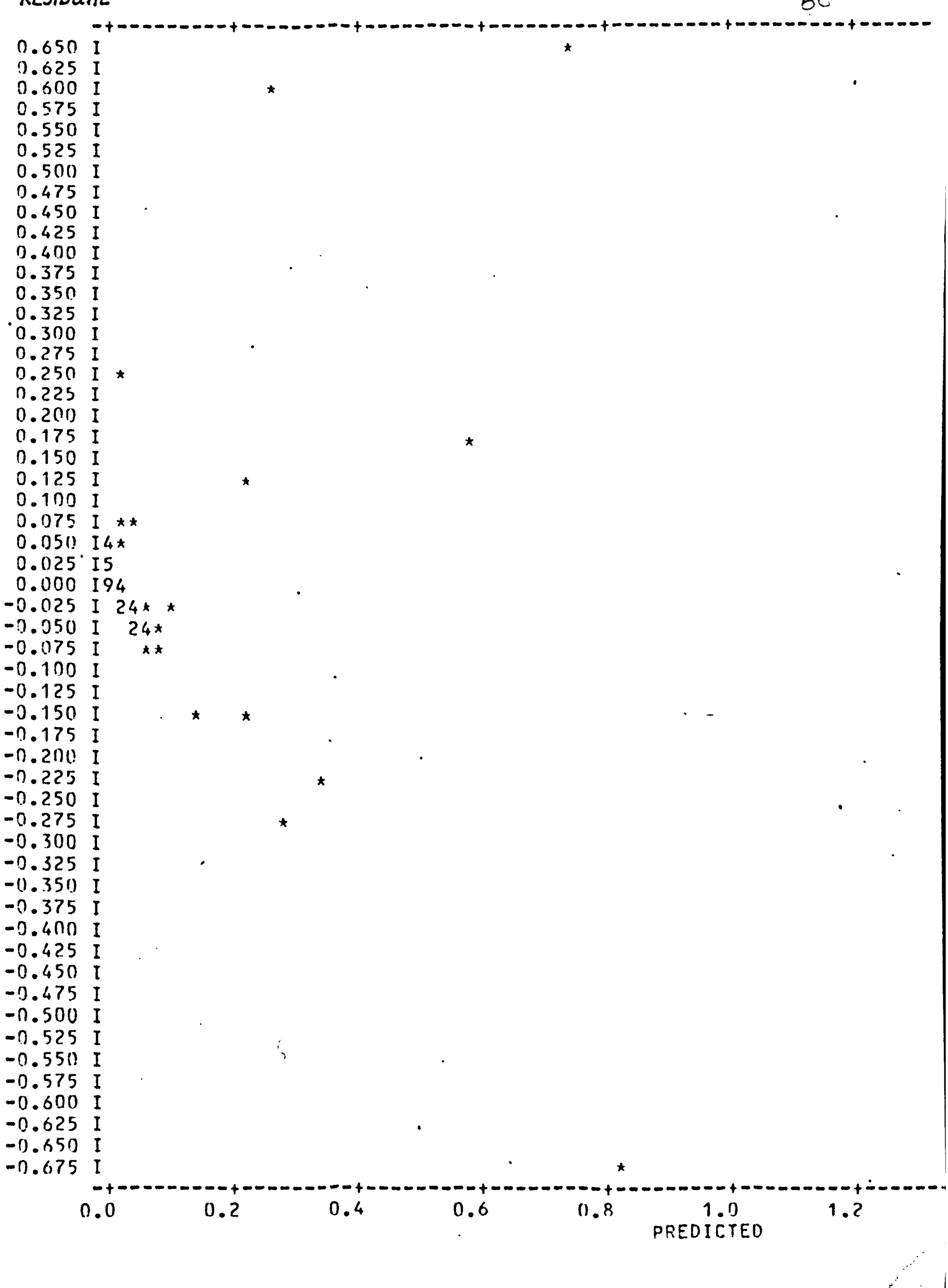
*** FITTED VALUES AND RESIDUALS ***

UNIT	OBSERVED	FITTED	RESIDUAL
1	0.0132	0.0083	0.0049
2	0.0127	0.0132	-0.0005
3	0.0319	0.0212	0.0107
4	0.0317	0.0072	0.0244
5	0.0090	0.0664	-0.0575
6	0.0029	0.0052	-0.0023
7	0.0244	0.0185	0.0058
8	0.0266	0.0075	0.0191
9	0.0026	0.0116	-0.0090
10	0.1012	0.0192	0.0820
11	0.0123	0.0388	-0.0265
12	0.2608	0.0198	0.2410
13	0.0740	0.0124	0.0616
14	0.0177	0.0048	0.0128
15	0.0411	0.0010	0.0402
16	0.0342	0.0077	0.0265
17	0.0096	0.0018	0.0078
18	0.0450	0.0008	0.0442
19	0.0661	0.0929	-0.0269
20	0.0232	0.0508	-0.0276
21	0.3469	0.2152	0.1316
22	0.8670	0.2546	0.6124
23	1.3880	0.7458	0.6422
24	0.7573	0.5701	0.1873
25	0.0177	0.0013	0.0164
26	0.0070	0.2816	-0.2746
27	0.0298	0.0869	-0.0571
28	0.0606	0.2226	-0.1620
29	0.1544	0.8249	-0.6705
30	0.1151	0.0351	0.0800
31	0.0076	0.0035	0.0042
32	1.6283	1.6149	0.0134
33	0.1196	0.3444	-0.2249
34	0.0660	0.0098	0.0562
35	0.0219	0.0401	-0.0183
36	0.0538	0.0000	0.0538
37	0.0000	0.0001	-0.0001
38	0.0000	0.0003	-0.0003
39	0.0000	0.0001	-0.0001
40	0.0000	0.0642	-0.0642
41	0.0000	0.0508	-0.0508

42	0.0000	0.0307	-0.0307
43	0.0000	0.0444	-0.0444
44	0.0000	0.0444	-0.0444
45	0.0000	0.0132	-0.0132
46	0.0000	0.0089	-0.0089
47	0.0000	0.0010	-0.0010
48	0.0000	0.0026	-0.0026
49	0.0000	0.0013	-0.0013
50	0.0000	0.0002	-0.0002
51	0.0000	0.0001	-0.0001
52	0.0000	0.0001	-0.0001
53	0.0000	0.0000	-0.0000
54	0.0000	0.0005	-0.0005
55	0.0000	0.0023	-0.0023
56	0.0000	0.0083	-0.0083
57	0.0000	0.0020	-0.0020
58	0.0000	0.0339	-0.0339
59	0.0000	0.0525	-0.0525
60	0.0000	0.0581	-0.0581
61	0.0000	0.0001	-0.0001
62	0.0000	0.0001	-0.0001
63	0.0000	0.1391	-0.1391
64	0.0000	0.0251	-0.0251
65	0.0000	0.0030	-0.0030
66	0.0000	0.0002	-0.0002
67	0.0000	0.0000	-0.0000
68	0.0000	0.0735	-0.0735
69	0.0000	0.0000	-0.0000
70	0.0000	0.0001	-0.0001

```
40  'VARIATE' YPRE $ ILEN
41  'CALC' YPRE=A*EXP(R*COST)
42  'HEAD' H1=''RESIDUAL''
43  'HEAD' H2=''PREDICTED''
44  'GRAPH/ATY=H1,ATX=H2' PES1 ; YPRE
45  'RUN'
```

RESIDUAL


0.650
0.625
0.600
0.575
0.550
0.525
0.500
0.475
0.450
0.425
0.400
0.375
0.350
0.325
0.300
0.275
0.250
0.225
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
-0.025
-0.050
-0.075
-0.100
-0.125
-0.150
-0.175
-0.200
-0.225
-0.250
-0.275
-0.300
-0.325
-0.350
-0.375
-0.400
-0.425
-0.450
-0.475
-0.500
-0.525
-0.550
-0.575
-0.600
-0.625
-0.650
-0.675
0.0
0.2
0.4
0.6
0.8
1.0
1.2
PREDICTED

```
46   '' STEP 2 USING VARIANCES FROM STEP 1 ''
47   'VARIATE' MR2 $ ILEN
48   'CALC' MR2=IZOP/(A*EXP(B*COST))
49   'RUN'


50   'VARIATE' MR3 $ ILEN
51   'CALC' MR3=MR2/MEAN(MR2)
52   'RUN'


53   'MODEL' MYFUN $ MR=A*EXP(B*COST)
54   'PRINT/P' COST,VISI,IZOP,LOG1,LOG2,MR2
55   'VARIATE' STEPL= 0.005 , 0.0005
56   'OPTIMIZE/NPAR=2,NT=MR3,LIK=2,PRINT=MPF,ITER=80'
57   MODEL=MYFUN;PARAM=A,B;STEPS=STEPL;Y=VISI;Z=MR;RES=RES1
58   'OUTPUT' 2
59   'PRINT/P' A,B $ 12.8 , 12.8
60   'OUTPUT' 1
61   'RUN'
```

COST	VISI	IZOP	LOG1	LOG2	MR2
3.6800E 1	1.3200E -2	2.2424E 4	F	T	2.7091E 6
3.4000E 1	1.2703E -2	3.9361E 4	F	T	2.9713E 6
3.1200E 1	3.1896E -2	2.9628E 4	T	T	1.3975E 6
3.7600E 1	3.1682E -2	4.7598E 4	F	T	6.5773E 6
2.4400E 1	8.9707E -3	4.3698E 4	T	T	6.5786E 5
3.9600E 1	2.8682E -3	5.2995E 4	F	T	1.0246E 7
3.2000E 1	2.4380E -2	2.6948E 4	F	T	1.4539E 6
3.7400E 1	2.6609E -2	1.0410E 4	T	T	1.3910E 6
3.4800E 1	2.5775E -3	2.7546E 4	T	T	2.3784E 6
3.1800E 1	1.0120E -1	1.3498E 4	T	T	7.0418E 5
2.7600E 1	1.2303E -2	1.7963E 4	T	T	4.6287E 5
3.1600E 1	2.6078E -1	1.2148E 4	T	T	6.1282E 5
3.4400E 1	7.3958E -2	1.1209E 4	F	T	9.0495E 5
4.0000E 1	1.7663E -2	6.7940E 3	T	T	1.4049E 6
4.9600E 1	4.1145E -2	4.4720E 3	T	T	4.6369E 6
3.7200E 1	3.4195E -2	2.3600E 4	T	T	3.0493E 6
4.6000E 1	9.5745E -3	6.5800E 3	T	T	3.7271E 6
5.1000E 1	4.4961E -2	8.3850E 3	T	T	1.0999E 7
2.2400E 1	6.6060E -2	2.2949E 4	T	T	2.4692E 5
2.6000E 1	2.3170E -2	1.6789E 4	T	T	3.3067E 5
1.7400E 1	3.4635E -1	1.3905E 4	T	T	6.4606E 4
1.6400E 1	8.6702E -1	5.3090E 3	T	T	2.0853E 4
1.0000E 1	1.3880E 0	1.2223E 4	T	T	1.6388E 4
1.1600E 1	7.5734E -1	4.2940E 3	T	T	7.5322E 3
4.8000E 1	1.7669E -2	1.6413E 4	F	T	1.3008E 7
1.5800E 1	6.9930E -3	1.8161E 4	T	T	6.4497E 4
2.2800E 1	2.9800E -2	1.0000E 4	F	T	1.1507E 5
1.7200E 1	6.0615E -2	1.7108E 4	T	T	7.6863E 4
9.4000E 0	1.5438E -1	7.4880E 3	T	T	9.0774E 3
2.8200E 1	1.1508E -1	1.0488E 4	F	T	2.9890E 5
4.2000E 1	7.6224E -3	1.1545E 4	F	F	3.3403E 6

5.4000E	0	1.6283E	0	6.3680E	3	T	T	3.9432E	3
1.4600E	1	1.1959E	-1	5.8200E	3	F	T	1.6397E	4
3.5800E	1	6.5975E	-2	3.7590E	3	F	T	3.8392E	5
2.7400E	1	2.1853E	-2	2.2880E	3	F	T	5.7009E	4
6.9600E	1	5.3794E	-2	1.0410E	3	F	F	3.1041E	7
6.6400E	1	0.0000E	0	3.5780E	3	F	F	6.2334E	7
5.6600E	1	0.0000E	0	2.3620E	3	F	T	7.9354E	6
6.1600E	1	0.0000E	0	5.7310E	3	F	F	4.4588E	7
2.4600E	1	0.0000E	0	8.4400E	3	F	T	1.3140E	5
2.6000E	1	0.0000E	0	6.0550E	3	F	T	1.1926E	5
2.9000E	1	0.0000E	0	3.1250E	3	F	F	1.0187E	5
2.6800E	1	0.0000E	0	1.0634E	4	F	T	2.3956E	5
2.6800E	1	0.0000E	0	2.0880E	3	F	T	4.7039E	4
3.4000E	1	0.0000E	0	8.0400E	2	F	T	6.0693E	4
3.6400E	1	0.0000E	0	2.6970E	3	F	F	3.0466E	5
4.9400E	1	0.0000E	0	2.1710E	3	F	F	2.1767E	6
4.3800E	1	0.0000E	0	6.4190E	3	F	F	2.5127E	6
4.8000E	1	0.0000E	0	3.5490E	3	F	F	2.8127E	6
5.7800E	1	0.0000E	0	1.5670E	3	F	F	6.4400E	6
6.1800E	1	0.0000E	0	3.3820E	3	F	F	2.7211E	7
6.4400E	1	0.0000E	0	3.8690E	3	F	F	4.8174E	7
6.9000E	1	0.0000E	0	4.4400E	2	F	F	1.1970E	7
5.3400E	1	0.0000E	0	1.1760E	3	F	F	2.3083E	6
4.4400E	1	0.0000E	0	3.7000E	2	F	F	1.6019E	5
3.6800E	1	0.0000E	0	2.3200E	2	F	F	2.8028E	4
4.5200E	1	0.0000E	0	1.1545E	4	F	F	5.7172E	6
2.8400E	1	0.0000E	0	5.6000E	3	F	F	1.6505E	5
2.5800E	1	0.0000E	0	7.7510E	3	F	T	1.4762E	5
2.5200E	1	0.0000E	0	4.5000E	1	F	F	7.7488E	2
6.3200E	1	0.0000E	0	3.9370E	3	F	F	4.0073E	7
6.4400E	1	0.0000E	0	5.3000E	2	F	F	6.5991E	6
2.0000E	1	0.0000E	0	3.5000E	1	F	T	2.5166E	2
3.0200E	1	0.0000E	0	1.4070E	3	F	F	5.6106E	4
4.2800E	1	0.0000E	0	2.4420E	3	F	F	8.0814E	5
5.7800E	1	0.0000E	0	5.5440E	3	F	F	2.2785E	7
6.9600E	1	0.0000E	0	2.3060E	3	F	F	6.8762E	7
2.3800E	1	0.0000E	0	7.3300E	2	F	T	9.9773E	3
6.8400E	1	0.0000E	0	6.3100E	2	F	F	1.5381E	7
6.3800E	1	0.0000E	0	2.7051E	4	F	F	3.0453E	8

56.....

*** CONVERGENCE MONITORING ***

ITER	FNO.	MOVE		FUNC VALUE	CURRENT PARAMETERS	
				STEPS	0.005000	0.000500
0	1	0		0.01861309	3.999686	-0.167947
				STEPS	0.011722	0.000213
1	12	5		0.01827950	2.770215	-0.143863
2	21	0		0.01818588	1.712788	-0.121681
3	30	0		0.01817914	1.831975	-0.123331
4	39	0		0.01817894	1.869615	-0.124093
				STEPS	0.002662	0.000059
5	48	1		0.01817894	1.870629	-0.124109
6	58	6		0.01817894	1.870202	-0.124099
				STEPS	0.144856	0.003194
1	67	0		0.01817894	1.870202	-0.124099

*** PARAMETER ESTIMATES ***

	ESTIMATE	S.E.	CORRELATIONS	
A	1.87020	1.39253	1.0000	
B	-0.12410	0.03060	-0.9581	1.0000

	DF	SS	MS
RESIDUAL	68	0.018179	0.000267

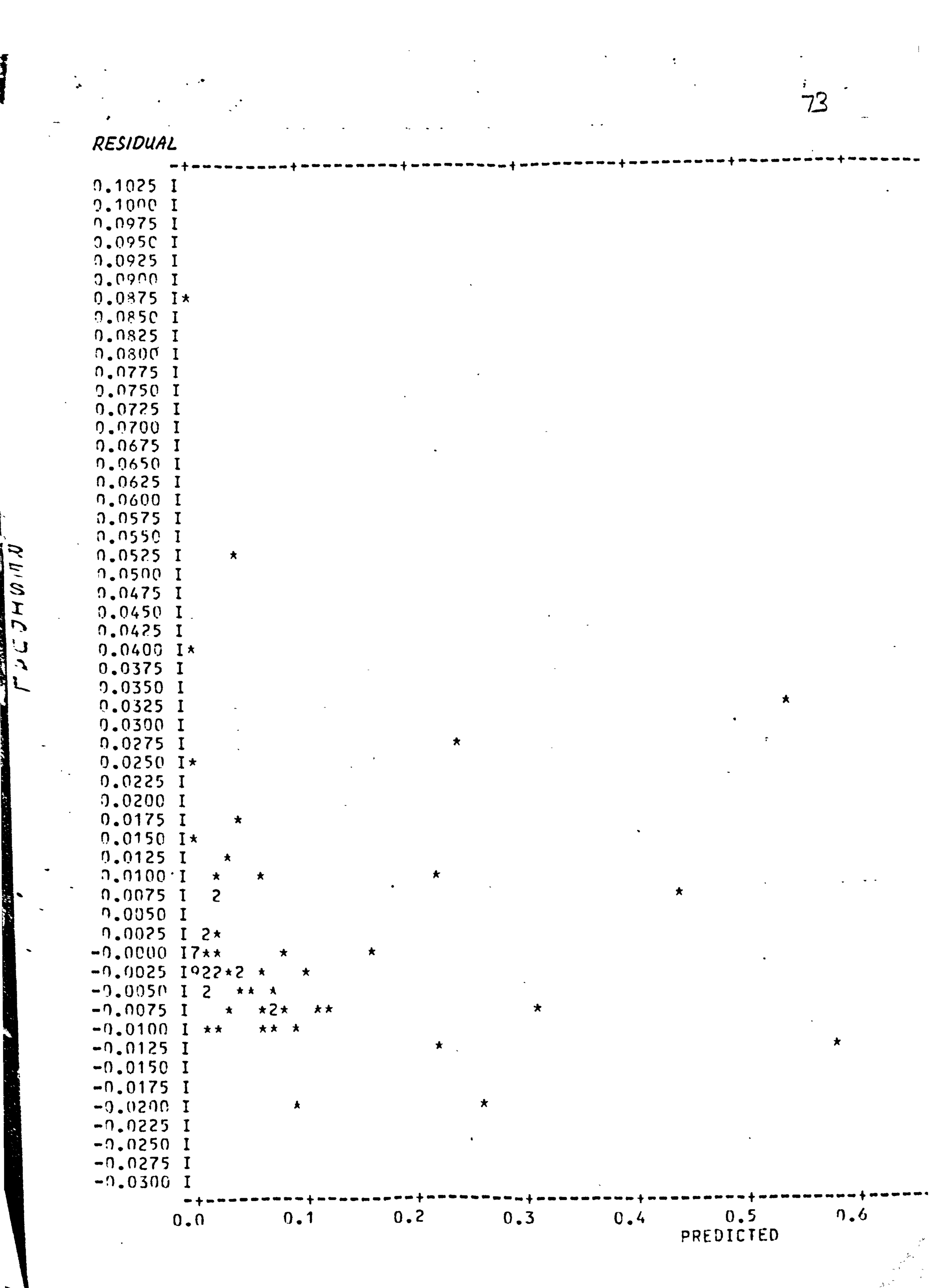
*** FITTED VALUES AND RESIDUALS ***

UNIT	OBSERVED	FITTED	RESIDUAL	WEIGHT
1	0.0132	0.0194	-0.0031	0.2400
2	0.0127	0.0275	-0.0076	0.2633
3	0.0319	0.0389	-0.0025	0.1238
4	0.0317	0.0176	0.0108	0.5827
5	0.0090	0.0905	-0.0197	0.0583
6	0.0029	0.0137	-0.0103	0.9078
7	0.0244	0.0353	-0.0039	0.1289
8	0.0266	0.0180	0.0030	0.1232
9	0.0026	0.0249	-0.0103	0.2107
10	0.1012	0.0361	0.0163	0.0624
11	0.0123	0.0609	-0.0098	0.0410
12	0.2608	0.0370	0.0521	0.0543
13	0.0740	0.0262	0.0135	0.0802
14	0.0177	0.0131	0.0016	0.1245
15	0.0411	0.0040	0.0238	0.4108
16	0.0342	0.0185	0.0082	0.2702
17	0.0096	0.0062	0.0019	0.3302
18	0.0450	0.0033	0.0411	0.9745
19	0.0661	0.1160	-0.0074	0.0219
20	0.0232	0.0742	-0.0087	0.0293
21	0.3469	0.2158	0.0099	0.0057
22	0.8670	0.2443	0.0268	0.0018
23	1.3880	0.5407	0.0323	0.0015
24	0.7573	0.4433	0.0081	0.0007
25	0.0177	0.0048	0.0138	1.1525
26	0.0070	0.2632	-0.0194	0.0057
27	0.0298	0.1104	-0.0081	0.0102
28	0.0606	0.2213	-0.0133	0.0068
29	0.1544	0.5825	-0.0121	0.0008
30	0.1151	0.0565	0.0095	0.0265
31	0.0076	0.0102	-0.0014	0.2959
32	1.6283	0.9569	0.0125	0.0003
33	0.1196	0.3055	-0.0072	0.0015
34	0.0660	0.0220	0.0081	0.0340
35	0.0219	0.0624	-0.0029	0.0051
36	0.0538	0.0003	0.0387	2.7502
37	0.0000	0.0005	-0.0012	5.5228
38	0.0000	0.0017	-0.0014	0.7031
39	0.0000	0.0009	-0.0018	3.9504
40	0.0000	0.0383	-0.0095	0.0116
41	0.0000	0.0742	-0.0076	0.0106

42	0.0000	0.0512	-0.0049	0.0090
43	0.0000	0.0672	-0.0098	0.0212
44	0.0000	0.0672	-0.0043	0.0042
45	0.0000	0.0275	-0.0020	0.0054
46	0.0000	0.0204	-0.0034	0.0270
47	0.0000	0.0041	-0.0018	0.1929
48	0.0000	0.0082	-0.0038	0.2226
49	0.0000	0.0048	-0.0024	0.2492
50	0.0000	0.0014	-0.0011	0.5706
51	0.0000	0.0009	-0.0014	2.4109
52	0.0000	0.0006	-0.0013	4.2681
53	0.0000	0.0004	-0.0004	1.0606
54	0.0000	0.0025	-0.0011	0.2045
55	0.0000	0.0076	-0.0009	0.0142
56	0.0000	0.0194	-0.0010	0.0025
57	0.0000	0.0069	-0.0049	0.5065
58	0.0000	0.0551	-0.0067	0.0146
59	0.0000	0.0761	-0.0087	0.0131
60	0.0000	0.0820	-0.0007	0.0001
61	0.0000	0.0007	-0.0014	3.5504
62	0.0000	0.0006	-0.0005	0.5847
63	0.0000	0.1563	-0.0007	0.0000
64	0.0000	0.0441	-0.0031	0.0050
65	0.0000	0.0092	-0.0025	0.0716
66	0.0000	0.0014	-0.0020	2.0187
67	0.0000	0.0003	-0.0008	6.0923
68	0.0000	0.0975	-0.0029	0.0009
69	0.0000	0.0004	-0.0004	1.3628
70	0.0000	0.0007	-0.0035	26.9810

```
62  'VARIATE' YPR2 $ ILEN
63  'CALC' YPR2=A*EXP(B*COST)
64  'GRAPH/ATY=H1,AIX=H2' RES1 ; YPR2
65  'RUN'
```

RESIDUAL


0.1025 I
0.1000 I
0.0975 I
0.0950 I
0.0925 I
0.0900 I
0.0875 I*
0.0850 I
0.0825 I
0.0800 I
0.0775 I
0.0750 I
0.0725 I
0.0700 I
0.0675 I
0.0650 I
0.0625 I
0.0600 I
0.0575 I
0.0550 I
0.0525 I
0.0500 I
0.0475 I
0.0450 I
0.0425 I
0.0400 I*
0.0375 I
0.0350 I
0.0325 I
0.0300 I
0.0275 I
0.0250 I*
0.0225 I
0.0200 I
0.0175 I
0.0150 I*
0.0125 I
0.0100 I
0.0075 I 2
0.0050 I
0.0025 I 2*
-0.0000 I7**
-0.0025 I922*2
-0.0050 I 2 **
-0.0075 I *2* **
-0.0100 I ** ** *
-0.0125 I
-0.0150 I
-0.0175 I
-0.0200 I
-0.0225 I
-0.0250 I
-0.0275 I
-0.0300 I
0.0 0.1 0.2 0.3 0.4 0.5 0.6
PREDICTED

```
   66  'CLOSE'
*****  END OF          EXPW01     AT LINE    56     USED 1204     LEFT 4796
```

NAME OF FOREST : HAPESKOVENE

01KRH 10 HARESKOVENE OG JONSTRUP VANG
8 1.32304 70 59104

PARAMETERS FROM 1st STAGE EXPFIT:

A1 = 3.99968560
A2 = -0.16794700

PARAMETERS FROM 2nd STAGE EXPFIT:

A2 = 1.87020180
B2 = -0.12409919

EXCLUDE IF LOG1 = A
EXCLUDE IF LOG2 = A

70 OBSERVATIONS OUT OF 70 SELECTED

GRIDFILE = HARE11.GRD

ZONE	NO OBS.	NO PRED.	CONS. SURPL.
1000	296	435	3511.
1500	500	1082	8724.
2000	945	1153	9296.
2100	1508	837	6749.
2200	392	3956	31881.
2300	152	727	5863.
2400	657	950	7656.
2450	277	187	1513.
2500	71	686	5529.
2600	1366	487	3931.
2610	221	1093	8810.
2620	3168	450	3627.
2630	829	293	2364.
2635	120	88	715.
2640	184	17	143.
2650	807	436	3517.
2660	63	40	329.
2670	377	27	225.
2700	1516	2663	21460.
2720	389	1246	10043.
2730	4823	3001	24183.
2740	4603	1297	10453.
2750	16966	6608	53253.
2760	3252	1903	15339.
2770	290	79	640.

2800	127	4780	38522.
2820	298	1104	8898.
2860	1037	3785	30501.
2880	1156	4361	35145.
2900	1207	592	4775.
3400	68	117	948.
3500	10369	6093	49101.
3520	696	1778	14327.
3650	248	82	666.
3660	50	142	1150.
4621	56	0	3.
2680	0	1	14.
2690	0	3	32.
2791	0	5	41.
2830	0	745	6007.
2840	0	449	3622.
2850	0	159	1288.
2920	0	714	5760.
2930	0	140	1131.
2942	0	22	178.
2950	0	55	444.
2960	0	8	71.
2970	0	52	422.
2980	0	17	138.
2990	0	2	18.
3050	0	2	24.
3060	0	2	20.
3080	0	0	1.
3320	0	2	23.
3330	0	2	23.
3340	0	4	36.
3400	0	79	637.
3450	0	308	2487.
3460	0	589	4753.
3471	0	3	30.
3480	0	2	23.
3490	0	0	3.
3530	0	5	44.
3540	0	62	500.
3550	0	22	182.
3600	0	7	64.
3630	0	0	6.
3670	0	71	576.
3310	0	0	2.
4000	0	18	149.

CONSUMERS SURPLUS EQUALS 452538.

PREDICTED NO. OF VISITORGROUPS 56160.
ACTUAL OBSERVED NO. OF VISITORGROUPS 59104