

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

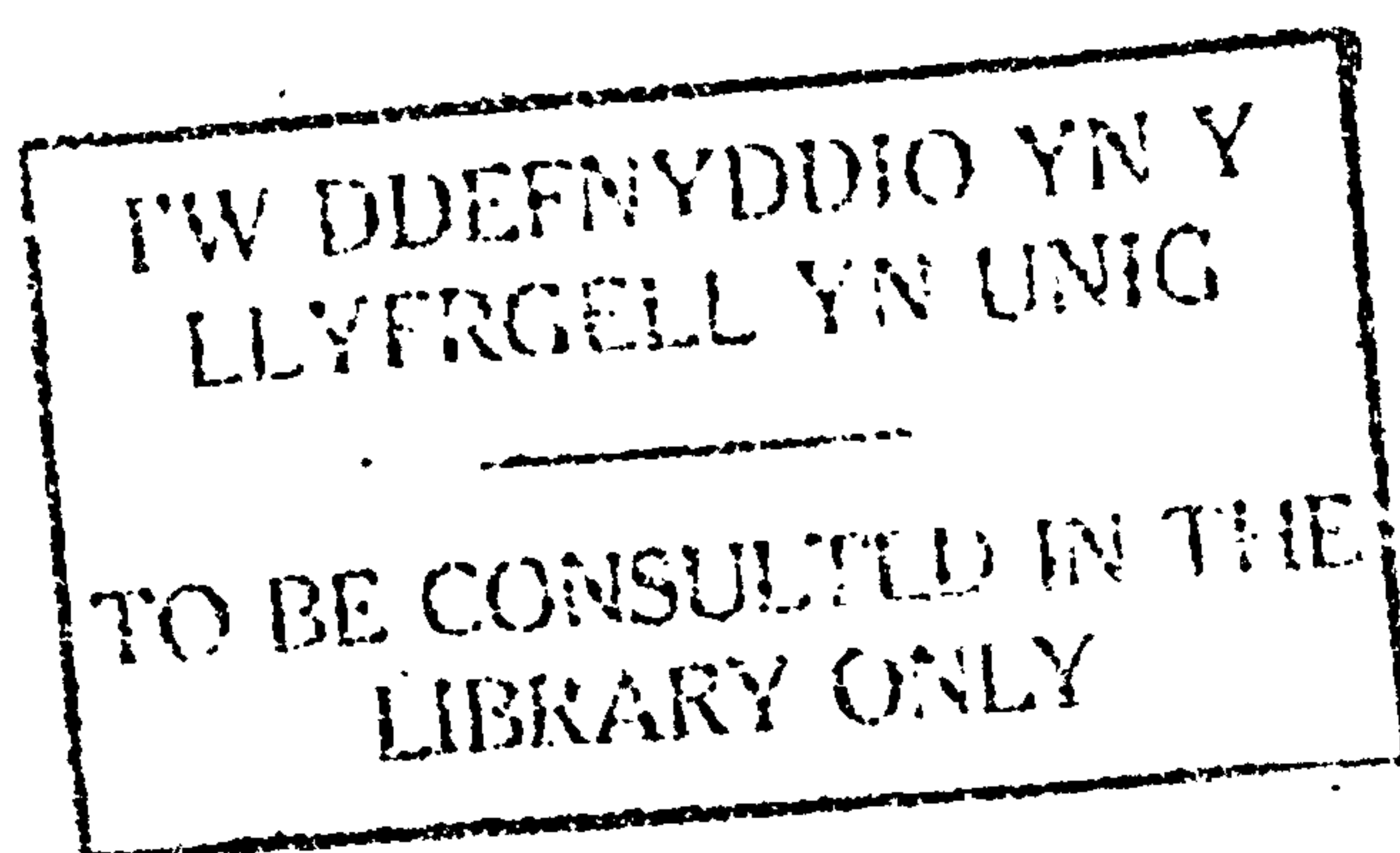
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

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 PFAF01.FOR
- A.5.3 Computer Programme PFAF02.FOR and PFAF03.FOR
- A.5.4 Example of Output



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


```

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 XPZG(21),YPRI(21),TIT1(10),TIT2(10),IZOP(21),
01300      1 IIZO(21),TIT3(10),XAR2(21),YAP2(21)
01400      DOUBLE PRECISION NAME1,NAME2,NAME3
01500      LOGICAL BATCH,MAT,SUBW
01600      COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT/VAR1/XRES(20),
01700      1 XRES2(20),XC2F(3)
01800      COMMON/WE1A/WEI1(21),WEI2(21),WEI3(21),WEI4(21),WEI5(21),WEI6(21)
01900      1 ,WEI7(21)
02000      CALL ERKSET(0)
02100      MAT=.FALSE.
02200      SUBW=.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)ANSW0
03700      300 FORMAT(A3)
03800      BATCH=.FALSE.
03900      IF (ANSW0.EQ.'YES') GO TO 700
04000      BATCH=.TRUE.
04100      310 WRITE(ITTY,400)
04200      400 FORMAT(/,4X,'INSTRUCTION FILE NAME ? ',S)
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) ? ',S)
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 ? ',S)
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='SERIN')
06200 ITTX=23
06300 790 WRITE(5,800)
06400 500 FORMAT(/,4X,'PLOTS WANTED (YES OR NO) ? ',5)
06500 READ(5,500,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(IITY,702)
07300 702 FORMAT(/,4X,'MATRIX AVAILABLE (YES OR NO) ? ',5)
07400 READ(ITIX,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(IITY,2110)
09900 2110 FORMAT(/,4X,'PLOT OF POINTS FOR TRIP DEMAND CURVE (YES OR NO) ? ',
10000 .15)
10100 7 READ(ITIX,300,ERR=6)ANSW4
10200 IF (ANSW4.EQ.'YES') IDUM=1
10300 2125 CONTINUE
10400 IF (IDUM.EQ.2) GO TO 2300
10500 WRITE(IITY,2200)
10600 2200 FORMAT(/,4X,'PLOT OF TRIP DEMAND CURVE:',/,4X,'*****')
10700 1*****',/)
10800 DATA I111/'POINTS FOR TRIP DEMAND CURVE*.'/
10900 NA1='A'
11000 DATA I113/'NO OF VISITORGROUPS'/'
11100 C
11200 COMMENT NON STANDARD FORTRAN
11300 C
11400 ENCODE(10,2250,TIT3(5))K1,NA1
11500 2250 FORMAT(18,A2)
11600 CALL PLOT1(1,XPZO,YPRI,IZON,2,XPZO,YPRI,IZON,TIT1,6,5,TIT3,6,5,1)
11700 DATA I112/'LINE THROUGH POINTS*.'/
11800 IFA12=1
11900 CALL SORT1(XPZO,YPRI,IZON,XAR2,YAR2,IFA12,IITY)
12000 CALL PLOT1(2,XPZO,YPRI,IZON,2,XAR2,YAR2,IZON,IIT1,6,5,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      2000 CONTINUE
13000      IDUM=2
13100      C
13200      C   FOR INTERACTIVE VERSION PROMPTS FOR KIND OF ANALYSIS
13300      C
13400      WRITE(ITY,3002)
13500      3002 FORMAT(/,4X,'TYPE OF ANALYSIS ? ',S)
13600      GO TO 3003
13700      2990 SKIP RECORD ITTX
13800      WRITE(ITY,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 SQR(ZPOP)/(NO OF VGROUPS)
14600      2)',//,6X,' 17: LINEAR REG. (WEIGHTED TO SQR(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 SQR(ZPOP)/(NO OF VGROUPS)
15300      2)',//,6X,' 27: 2ND DG POL. (WEIGHTED TO SQR(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 SQR(ZPOP)/(NO OF VGRO
16100      2UPS)',//
16200      2      ,6X,' 37: TRNF. LIN. REG. (WEIGHTED TO SQR(ZONEPOP))',//
16300      8      ,6X,' 4  : NO MORE ANALYSES',//
16400      9      ,52X,' ? ',S)
16500      3003 READ(ITTX,3004,ERR=2990)INAN
16600      3004 FORMAT(15)
16700      IF ((INAN.LT.1).OR.(INAN.GT.39)) GO TO 3018
16800      IF (INAN.EQ.4) GO TO 3012
16900      3005 WRITE(ITY,3010)
17000      3010 FORMAT(/,4X,'PLOTS WANTED (YES OR NO) ? ',S)
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(ITY,3020)
17900      3020 FORMAT(/,4X,'WRONG NUMBER OF ANALYSIS !!!')
18000      GOTJ 2900

```

```

78100 C
78200 C LINEAR REGRESSION
78300 C
78400 3100 CALL GLM2(1,YPRI,XPZO,IZON,WEI1)
78500 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
78600 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
78700 1,ITZO,ITOT)
78800 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
78900 IF (.NOT.BATCH) GO TO 2900
79000 C
79100 C WEIGHTED LINEAR REGRESSION (ACCORDING TO NO OF VISITORGROUPS)
79200 C
79300 3200 CALL GLM2(2,YPRI,XPZO,IZON,WEI4)
79400 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
79500 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
79600 1,ITZO,ITOT)
79700 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
79800 IF (.NOT.BATCH) GO TO 2900
79900 C
20000 C
20100 C WEIGHTED LINEAR REGRESSION (ACCORDING TO SQRT(ZONEPOPULATION) )
20200 C
20300 3210 CALL GLM2(3,YPRI,XPZO,IZON,WEI2)
20400 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
20500 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
20600 1,ITZO,ITOT)
20700 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
20800 IF (.NOT.BATCH) GO TO 2900
20900 C
21000 C WEIGHTED LINEAR REGRESSION (SQRT(ZONEPOP)*NO OF VISITORGROUPS)
21100 C
21200 3220 CALL GLM2(4,YPRI,XPZO,IZON,WEI5)
21300 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
21400 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
21500 1,ITZO,ITOT)
21600 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
21700 IF (.NOT.BATCH) GO TO 2900
21800 C
21900 C WEIGHTED LINEAR REGRESSION (ZONEPOP/VISITRATE)
22000 C
22100 3230 CALL GLM2(5,YPRI,XPZO,IZON,WEI3)
22200 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
22300 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
22400 1,ITZO,ITOT)
22500 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
22600 IF (.NOT.BATCH) GO TO 2900
22700 IF (BATCH) GO TO 3290
22800 C
22900 C WEIGHTED LINEAR REGRESSION (SUBJECTIVE)
23000 C
23100 3240 CALL SUBWEI(IZON,WEI7,SUBW)
23200 CALL GLM2(6,YPRI,XPZO,IZON,WEI7)
23300 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
23400 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
23500 1,ITZO,ITOT)
23600 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
23700 IF (.NOT.BATCH) GO TO 2900
23800 3290 CONTINUE
23900 C
24000 C WEIGHTED LINEAR REGRESSION (ACCORDING TO ZONEPOPULATION )

```



```

24100 3250 CALL CLR2(7,YPRI,XPZO,IZON,WEI6)
24200 CALL AGGR2(1,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
24300 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,1,XPZO,YPRI,IZON,IZOP
24400 1,ITZO,ITOT)
24500 CALL AGGR2(3,1,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
24600 IF (.NOT.BATCH) GO TO 2900
24700 C
24800 C 2ND DG POLYNOMIAL
24900 C
25000 3300 CALL LSF12(1,YPRI,XPZO,IZON,WEI1)
25100 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
25200 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
25300 1,ITZO,ITOT)
25400 CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
25500 IF (.NOT.BATCH) GO TO 2900
25600 C
25700 C WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO NO OF VISITORS)
25800 C
25900 3400 CALL LSF12(2,YPRI,XPZO,IZON,WEI4)
26000 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
26100 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
26200 1,ITZO,ITOT)
26300 CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
26400 IF (.NOT.BATCH) GO TO 2900
26500 C
26600 C WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO SQRT(ZONEPOPULATION) )
26700 C
26800 3410 CALL LSF12(3,YPRI,XPZO,IZON,WEI2)
26900 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
27000 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
27100 1,ITZO,ITOT)
27200 CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
27300 IF (.NOT.BATCH) GO TO 2900
27400 C
27500 C WEIGHTED 2ND DG POLYNOMIAL ( SQRT(ZONEPOP)*NO OF VISITORGROUPS )
27600 C
27700 3420 CALL LSF12(4,YPRI,XPZO,IZON,WEI5)
27800 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
27900 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
28000 1,ITZO,ITOT)
28100 CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
28200 IF (.NOT.BATCH) GO TO 2900
28300 C
28400 C WEIGHTED 2ND DG POLYNOMIAL (ACCORDING TO ZONEPOP/VISITRATE)
28500 C
28600 3430 CALL LSF12(5,YPRI,XPZO,IZON,WEI3)
28700 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
28800 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
28900 1,ITZO,ITOT)
29000 CALL AGGR2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
29100 IF (.NOT.BATCH) GO TO 2900
29200 IF (BATCH) GO TO 3490
29300 C
29400 C WEIGHTED 2ND DG POLYNOMIAL (SUBJECTIVE)
29500 C
29600 3440 CALL SUBWEI(IZON,WEI7,SUBW)
29700 CALL LSF12(6,YPRI,XPZO,IZON,WEI7)
29800 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
29900 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,2,XPZO,YPRI,IZON,IZOP
30000 1,ITZO,ITOT)

```

```

30100 CALL AGGP2(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 LSF12(7,YPRI,XPZO,IZON,WEI6)
30800 CALL AGGR2(1,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
30900 IF ((.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGP2(2,2,XPZO,YPRI,IZON,IZOP
31000 1,ITZO,ITOT)
31100 CALL AGGP2(3,2,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
31200 IF (.NOT.BATCH) GO TO 2900
31300 C
31400 C TRNF LIN. REG.
31500 C
31600 3500 CALL TRNF(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 AGGP2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
32100 IF (.NOT.BATCH) GO TO 2900
32200 C
32300 C WEIGHTED TRNF LIN. REG. (ACCORDING TO NO OF VISITORGROUPS)
32400 C
32500 3600 CALL TRNF(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 TRNF LIN. REG. (ACCORDING TO SQRT(ZONEPOPULATION))
33300 C
33400 3610 CALL TRNF(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 TRNF LIN. REG. (SQRT(ZONEPOP)*NO OF VISITORGROUPS)
34200 C
34300 3420 CALL TRNF(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 TRNF LIN. REG. (ZONEPOP/VISITRATE)
35100 C
35200 3630 CALL TRNF(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)

```



```

36100 C
36200 3640 CALL SUBREI(IZON,WEI7,SUBN)
36300 CALL TRNF(6,YPRI,XPZO,IZON,WEI7)
36400 CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
36500 IF (.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
36600 1,ITZO,ITOT)
36700 CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
36800 IF (.NOT.BATCH) GO TO 2900
36900 3690 CONTINUE
37000 C
37100 C WEIGHTED TRNF LIN. REG. (ACCORDING TO ZONEPOPULATION)
37200 C
37300 3650 CALL TRNF(7,YPRI,XPZO,IZON,WEI6)
37400 CALL AGGR2(1,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
37500 IF (.NOT.TRAD).AND.(.NOT.MAT)) CALL AGGR2(2,3,XPZO,YPRI,IZON,IZOP
37600 1,ITZO,ITOT)
37700 CALL AGGR2(3,3,XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
37800 IF (.NOT.BATCH) GO TO 2900
37900 C
38000 C TERMINATING THE PROGRAMME
38100 C
38200 IF (BATCH) GO TO 4030
38300 4000 CONTINUE
38400 IF (.NOT.BATCH) IDUM=2
38500 GO TO 4008
38600 4005 SKIP RECORD ITTX
38700 4008 WRITE(IITY,4010)
38800 4010 FORMAT(/,4X,'OPTIONS : ',/
38900 1 ,4X,'1: BEGIN WITH NEW DATA ',/
39000 2 ,4X,'2: EXIT FROM PROGRAMME',/
39100 3 ,30X,'? ',8)
39200 READ(ITTX,*,ERR=4005)IANS
39300 GO TO (110,4030)IANS
39400 GO TO 2900
39500 4030 CONTINUE
39600 C
39700 C CLOSE AND DELETE TMP-FILE
39800 C
39900 WRITE(IITY,2800)
40000 2800 FORMAT(///,4X,'EXIT FROM CLAWSON PROGRAMME',//,4X,'C*jbch1981',//)
40100 CLOSE(UNIT=22,FILE=NAME3,DISPOSE='DELETE')
40200 IF (BATCH) CLOSE(UNIT=23,FILE=NAME1,DISPOSE='SAVE')
40300 IF (BATCH) CLOSE(UNIT=24,FILE=NAME2,DISPOSE='SAVE')
40400 STOP
40500 END

```



```

00100 C ***** SUB15.FOR *****
00200 C
00300 C
00400 C SURROUTINE INOUT MATIN
00500 C
00600 C
00700 C SURROUTINE INOUT(XPZO,YPRI,IZON,IZOP,ITZO,ITOT,BATCH,NAME3)
00800 COMMENT
00900 C
01000 C SUBROUTINE INOUT
01100 C
01200 C XPZO: NO/K1 VISITING
01300 C YPRI: PRICE FROM ZONE (AVERAGE OR FIXED)
01400 C IZON: NO OF ZONES
01500 C IZOP: POPULATION IN ZONE
01600 C ITZO: NO OF VISITORGROUPS FROM ZONE
01700 C BATCH : LOGICAL
01800 C
01900 C SEE DIMENSION STATEMENT FOR ARRAYS
02000 COMMENT
02100 DIMENSION NAM(10),X(10),IZOP(21),ITZO(21),IZON(21),
02200 1 XPZO(21),YPRI(21),YPR2(21),IDEN(21),XNEW(21),YPR3(21)
02300 COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CUNT
02400 DOUBLE PRECISION NAME1,XPZO1(21),XMAX,XMIN,NAME3,NAME2
02500 LOGICAL BATCH,TRAD,NSUC,EZON,RWEI
02600 140 CONTINUE
02700 TRAD=.FALSE.
02800 EZON=.FALSE.
02900 RWEI=.FALSE.
03000 COMMENT
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 CLAWSON',/
04000 2 6X,' 2: REVISED CLAWSON',/
04100 3 6X,' 3: REV. CLAWSON (WEIGH.)',/
04200 4 6X,' ? ',S)
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,' ? ',S)
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 ? ',S)
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) ? ',S)
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,'? ',S)
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 ? ',S)
08900      READ(ITIX,*,ERR=470)NUM0
09000 480  FORMAT(4X,I6,/)
09100      IF (BATCH) WRITE(ITTY,480)NUM0
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) ? ',S)
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 490  CONTINUE
11900 C
12000 C  INITIALIZES THE VARIABLES

```



```

12100
12200
12300 REMIND 20
12400 DO 150 I=1,21
12500 IT7J(I)=0.0
12600 YPR1(I)=0.0
12700 YPR3(I)=0.0
12800 XFR(I)=0.0
12900 IUEV(I)=0
13000 ITZJ(I)=0
13100 IZOS(I)=0
13200 IZOP(I)=0
13300 150 CONTINUE
13400 ITOT=0
13500 GO TO 490
13600 430 SKIP RECORD ITTX
13700 490 WRITE(ITTY,500)
13800 500 FORMAT(4X,'HOW MANY ZONES (MAX 20) ? ',S)
13900 READ(ITIX,*,ERR=488)IZON
14000 600 FORMAT(4X,I6,/)
14100 IF (BATCH) WRITE(ITTY,600)IZON
14200 IF (IZON.LE.0) GOTO 605
14300 DO 605 I=1,IZON
14400 IF (E2CN) GO TO 650
14500 IF (I.EQ.IZON) GO TO 609
14600 602 CONTINUE
14700 GO TO 592
14800 591 SKIP RECORD ITTX
14900 592 WRITE(ITTY,601)I
15000 601 FORMAT(4X,'ZONE NO ',I2,' EXTENDS TO MILES ? ',S)
15100 READ(ITIX,*,ERR=591)IZOB(I)
15200 IF (BATCH) WRITE(ITTY,604)IZOB(I)
15300 606 CONTINUE
15400 GO TO 594
15500 593 SKIP RECORD ITTX
15600 594 WRITE(ITTY,603)I
15700 603 FORMAT(4X,'POPULATION IN ZONE NO ',I2,' ? ',S)
15800 READ(ITIX,*,ERR=593)IZOP(I)
15900 604 FORMAT(4X,I10,/)
16000 IF (BATCH) WRITE(ITTY,604)IZOP(I)
16100 IF (.NOT. IRAD) GO TO 605
16200 GO TO 596
16300 595 SKIP RECORD ITTX
16400 596 WRITE(ITTY,616)I
16500 616 FORMAT(4X,'TRAVEL COST FROM ZONE NO ',I2,' (PENNY) ? ',S)
16600 READ(ITIX,*,ERR=595)YPR2(I)
16700 IF (BATCH) WRITE(ITTY,618)YPR2(I)
16800 618 FORMAT(8X,F6.0,/)
16900 605 CONTINUE
17000 GO TO 650
17100 609 WRITE(ITTY,610)
17200 610 FORMAT(4X,'DOES THE OUTER ZONE HAVE ANY BOUNDARY (YES OR NO) ? ',S)
17300 1)
17400 READ(ITIX,225)ANSW0
17500 615 FORMAT(4X,A3,/)
17600 IF (BATCH) WRITE(ITTY,615)ANSW0
17700 IF (ANSW0.EQ.'YES') GO TO 602
17800 IZOB(IZON)=10000
17900 GO TO 606
18000 650 CONTINUE
GO TO 652

```

```

18100 631 SKIP RECORD ITTX
18200 632 WRITE(ITTY,652)I
18300 652 FORMAT(4X,"ZONE NO ",12," HAS IDENTIFICATION NO ? ",5)
18400 READ(ITIX,*,ERR=651)IDEN(I)
18500 IF (BATCH) WRITE(ITTY,653)IDEN(I)
18600 653 FORMAT(4X,15,/)
18700 GO TO 606
18800 630 CONTINUE
18900 C
19000 C SUMMARY OF INPUT ON ZONES
19100 C
19200 WRITE(ITTY,640)
19300 640 FORMAT(1H1,/,5X,"SUMMARY OF INPUT:",/,5X,"*****",/)
19400 DO 4005 I=1,IZON
19500 IF (TRAD) GO TO 4007
19600 IF (EZON) GO TO 646
19700 WRITE(ITTY,4002)I,IZON(I),IZOP(I)
19800 4002 FORMAT(8X,"ZONE NO ",12," UNTIL ",15," MILES ; POPULATION ",18)
19900 GO TO 4008
20000 646 CONTINUE
20100 WRITE(ITTY,648)I,IDEN(I),IZOP(I)
20200 648 FORMAT(6X,"ZONE NO ",12," ; IDENTIFICATION NO ",15," ; POPULATION ",
20300 1 15)
20400 GO TO 4008
20500 4007 CONTINUE
20600 IF (EZON) GO TO 660
20700 WRITE(ITTY,4006)I,IZON(I),IZOP(I),YPR2(I)
20800 4006 FORMAT(8X,"ZONE NO ",12," UNTIL ",15," MILES ; POPULATION ",18,
20900 1 " ; TRAVEL COSTS",F8.0)
21000 GO TO 4008
21100 660 CONTINUE
21200 WRITE(ITTY,662)I,IDEN(I),IZOP(I),YPR2(I)
21300 662 FORMAT(8X,"ZONE NO ",12," ; IDENT.NO ",15," ; POPUL. ",18,
21400 1 " ; TR.COST ",F8.0)
21500 4008 CONTINUE
21600 WRITE(ITTY,670)
21700 670 FORMAT(/,4X,"ANY CHANGES IN THE ZONING (YES OR NO) ? ",5)
21800 READ(ITIX,225)ANS#3
21900 671 FORMAT(4X,A3,/)
22000 IF (BATCH) WRITE(ITTY,671)ANS#3
22100 IF (ANS#3.EQ."YES") GO TO 498
22200 COMMENT
22300 C
22400 C CALCULATING % FROM EACH ZONE (XPZO,YPRI)
22500 C
22600 COMMENT
22700 IF (.NOT.EZON) GO TO 4090
22800 C
22900 C IDENTIFICATION OF VAR ON RAW DATA FILE
23000 C
23100 GO TO 4079
23200 4078 SKIP RECORD ITTX
23300 4079 WRITE(ITTY,4080)
23400 4080 FORMAT(4X,"ZONE IDENTIFIER IS VARIABLE NO ? ",5)
23500 READ(ITIX,*,ERR=4078)KST
23600 IF (BATCH) WRITE(ITTY,4052)KST
23700 GO TO 4092
23800 4090 CONTINUE
23900 GO TO 4091
24000 4089 SKIP RECORD ITTX

```



```

24100 4001 WRITE(ITTY,4050)
24200 4050 FORMAT(4X,'MINEDISTANCE IS VARIABLE NO ? ',S)
24300 READ(ITIX,*,ERR=4059)IST
24400 IF (BATCH) WRITE(ITTY,4052)IST
24500 4092 CONTINUE
24600 IF (TRAD) GO TO 718
24700 GO TO 4059
24800 4058 SKIP RECORD ITTX
24900 4059 WRITE(ITTY,4060)
25000 4060 FORMAT(4X,'TRAVELCOSTS IS VARIABLE NO. ? ',S)
25100 READ(ITIX,*,ERR=4058)JST
25200 4052 FORMAT(4X,12,/)
25300 IF (BATCH) WRITE(ITTY,4052)JST
25400 4037 GO TO 4039
25500 4038 SKIP RECORD ITTX
25600 4039 IF (.NOT.RWEI) GO TO 718
25700 WRITE(ITTY,4040)
25800 4040 FORMAT(4X,'STAYLENGTH IS VARIABLE NO. ? ',S)
25900 READ(ITIX,*,ERR=4038)IWE
26000 IF (BATCH) WRITE(ITTY,4052)IWE
26100 718 CONTINUE
26200 ANSW5='NO'
26300 IF ((ANS5#0.NE.'YES').OR.TRAD).OR.EZON) GO TO 730
26400 C
26500 C OBSERVATIONS OUTSIDE ZONES
26600 C
26700 WRITE(ITTY,720)
26800 720 FORMAT(1,4X,'ARE OBSERVATIONS OUTSIDE THE OUTER ZONE TO BE INCLUDE
26900 10 (YES OR NO) ? ',S)
27000 READ(ITIX,225)ANSW5
27100 IF (BATCH) WRITE(ITTY,671)ANSW5
27200 730 CONTINUE
27300 C
27400 C OPEN UNIT 22 AS WORKFILE FOR INDIVIDUAL INTEGRATION
27500 C
27600 CALL KANNA(1,NAME3)
27700 OPEN(UNIT=22,FILE=NAME3)
27800 DO 4200 I=1,NUMD
27900 READ(20,1000,ERR=733)(X(K),K=1,10)
28000 1000 FORMAT(10F)
28100 GO TO 734
28200 733 WRITE(ITTY,732)
28300 732 FORMAT(1,4X,'INPUT ERROR IN RAW DATA FILE ! ',/
28400 1,20X,' START AGAIN WITH INPUT ',/)
28500 GO TO 140
28600 734 CONTINUE
28700 IF (.NOT.EZON) GO TO 4140
28800 NSUC=.TRUE.
28900 DO 4120 J=1,IZON
29000 K5=J
29100 IF (.NOT.(X(K5T).EQ.IDEN(J))) GO TO 4120
29200 NSUC=.FALSE.
29300 GO TO 4122
29400 4120 CONTINUE
29500 4122 CONTINUE
29600 IF (NSUC) K5=K5+1
29700 ITZ0(K5)=ITZ0(K5)+1
29800 IF (.NOT.TRAD) YPRI(K5)=YPRI(K5)+X(JST)
29900 IF ((.NOT.TRAD).AND.(RWEI)) XNEW(K5)=XNEW(K5)+(1/X(IWE))
30000 IF ((.NOT.TRAD).AND.(R#EI)) YPR3(K5)=YPR3(K5)+(X(JST)/X(IWE))

```

```

30100 GO TO 4190
30200 4140 CONTINUE
30300 DO 4150 J=IZON,1,-1
30400 K5=J
30500 IF (X(JST).GT.IZOZ(J)) GO TO 4170
30600 4150 CONTINUE
30700 ITZJ(K5)=ITZJ(K5)+1
30800 IF (.NOT.IRAD) YPRI(K5)=YPRI(K5)+X(JST)
30900 IF (.NOT.TRAD).AND.(RNEI) XNEW(K5)=XNEW(K5)+(1/X(IWE))
31000 IF (.NOT.TRAD).AND.(RNEI) YPR3(K5)=YPR3(K5)+(X(JST)/X(IWE))
31100 GO TO 4190
31200 4170 ITZJ(K5+1)=ITZJ(K5+1)+1
31300 IF (.NOT.IRAD) YPRI(K5+1)=YPRI(K5+1)+X(JST)
31400 IF (.NOT.TRAD).AND.(RNEI) XNEW(K5+1)=XNEW(K5+1)+(1/X(IWE))
31500 IF (.NOT.TRAD).AND.(RNEI) YPR3(K5+1)=YPR3(K5+1)+(X(JST)/X(IWE))
31600 4190 CONTINUE
31700 IF (.NOT.TRAD).AND.((K5.LT.IZON).OR.((K5.EQ.IZON).AND.((ANSW5.EQ.
31800 1 'YES').OR.(EZON)))) WRITE(22,4194)X(JST)
31900 4194 FORMAT(F10.2)
32000 4200 CONTINUE
32100 IZOP=IZON+1
32200 DO 6000 I=1,IZON
32300 XPZU1(I)=DBLE(FLOAT(ITZO(I)))/FLOAT(IZOP(I))
32400 6000 CONTINUE
32500 XMIN=XPZU1(1)
32600 XMAX=XPZU1(1)
32700 DO 6010 I=1,IZON
32800 XMIN=DMIN(XMIN,XPZU1(I))
32900 XMAX=DMAX(XMAX,XPZU1(I))
33000 C
33100 C FINDS THE K1 VALUE ( NO/K1 INHABITANTS )
33200 C
33300 6010 CONTINUE
33400 K1=INT(10.0/XMAX)
33500 K3=1
33600 6020 CONTINUE
33700 K2=(K1/(10**K3))
33800 IF (K2.LE.0) GO TO 6050
33900 K3=K3+1
34000 GO TO 6020
34100 6030 CONTINUE
34200 K1=(10**K3)
34300 C
34400 C OUTPUT RESULTS ( XPZU YPRI )
34500 C
34600 WRITE(IITY,4053)
34700 WRITE(IITY,4054)K1
34800 4053 FORMAT(I11,/,5X,'RESULTS:',/,5X,'*****')
34900 4054 FORMAT(/,10X,'NO. OF VISITORGROUPS /',I10,' INHABITANTS IN ZONE : '
35000 1 ,/)
35100 DO 4300 I=1,IZON
35200 XPZU(I)=SNGL(XPZU1(I)*K1)
35300 IF (TRAD) GO TO 6055
35400 IF (ITZU(I).NE.0.0) YPRI(I)=YPRI(I)/FLOAT(ITZO(I))
35500 IF ((RNEI).AND.(ITZU(I).NE.0.0)) YPRI(I)=YPR3(I)/XNEW(I)
35600 6055 CONTINUE
35700 IF (.NOT.IRAD) GO TO 4260
35800 YPRI(I)=YPR2(I)
35900 WRITE(IITY,4252)I,XPZU(I),YPRI(I)
36000 4252 FORMAT(10X,'ZONE NO. ',I2,4X,F8.4,' ; TRAVEL COSTS ',F8.0)

```



```

36100      GO TO 4300
36200 4260 CONTINUE
36300      IF (PARE1) GO TO 4270
36400      WRITE(ITTY,4250)I,XPZO(I),YPRI(I)
36500 4250 FORMAT(10X,'ZONE NO ',I2,4X,F8.4,'      ; AVERAGE COST ',F8.2)
36600      GO TO 4300
36700 4270 CONTINUE
36800      WRITE(ITTY,4272)I,XPZO(I),YPRI(I)
36900 4272 FORMAT(10X,'ZONE NO. ',I2,4X,F8.4,'      ; AVERAGE COST (WEIGH) ',
37000      1F8.2)
37100 4300 CONTINUE
37200      XPCI=FLOAT(ITZO(IZON1))/FLOAT(NUMO)*100
37300 C
37400 C      OUTPUTS RESULTS OF OBSERVATIONS OUTSIDE ZONES
37500 C
37600      IF (ANSR5.EQ.'YES') GO TO 740
37700      IF (ITZO(IZON1).EQ.0) GO TO 4380
37800      WRITE(ITTY,4360)ITZO(IZON1),XPCT
37900 4360 FORMAT(/,6X,'(',I4,' OBSERVATIONS, OR ',F5.2,' PCT. OF ALL OBSERVA
38000      TIONS ARE OUTSIDE',/,10X,' THE OUTER ZONE, AND ARE GOING TO BE INC
38100      LUDED AS POINT:)',/)
38200      XPZO(IZON1)=0.0
38300      YPRI(IZON1)=YPRI(IZON1)/FLOAT(ITZO(IZON1))
38400      IF (RARE1) YPRI(IZON1)=YPR3(IZON1)/XNEW(IZON1)
38500      WRITE(ITTY,4250)IZON1,XPZO(IZON1),YPRI(IZON1)
38600 4380 CONTINUE
38700      WRITE(ITTY,4385)
38800 4385 FORMAT(X)
38900      IZON=IZON1
39000      GO TO 844
39100 740 CONTINUE
39200      IF (ITZO(IZON1).EQ.0) GO TO 844
39300      IF (EZON) GO TO 840
39400      WRITE(ITTY,4362)ITZO(IZON1),XPCT
39500 4362 FORMAT(/,6X,'(',I4,' OBSERVATIONS, OR ',F5.2,' PCT. OF ALL OBSERVA
39600      TIONS ARE OUTSIDE',/,10X,' THE OUTER ZONE, AND ARE EXCLUDED )')
39700      GO TO 844
39800 840 CONTINUE
39900      WRITE(ITTY,842)ITZO(IZON1),XPCT
40000 842 FORMAT(/,6X,'(',I4,' OBSERVATIONS, OR ',F5.2,
40100      1 ' PCT. OF ALL OBSERVATIONS HAD AN UNDEFINED',/,10X,
40200      2 ' ZONE IDENTIFIER, AND ARE EXCLUDED ! )')
40300 844 CONTINUE
40400      WRITE(ITTY,845)
40500 845 FORMAT(X)
40600      DO 4400 I=1,IZON
40700      WRITE(ITTY,4390)I,ITZO(I)
40800 4390 FORMAT(10X,'NUMBER FROM ZONE ',I2,4X,';',I6,' VISITORGROUPS')
40900      ITOT=ITOT+ITZO(I)
41000 4400 CONTINUE
41100      WRITE(ITTY,680)
41200 680 FORMAT(/,4X,'ANY CHANGES IN THE ZONING BEFORE FITTING THE CURVE (Y
41300      1ES OR NO) ? ',5)
41400      READ(ITIX,225)ANSW4
41500      IF (BATCH) WRITE(ITTY,671)ANSW4
41600      IF (ANSW4.EQ.'YES') GO TO 498
41700      CLOSE(UNIT=20,FILE=NAME1,DISPOSE='SAVE')
41800      WRITE(ITTY,6300)
41900 6300 FORMAT(/,4X,'INFORMATION TO BE STORED IN MATRIX (YES OR NO) ? ',5
42000      1)

```

```

42100      READ(ITIX,225)ANS..6
42200      IF (BATCH) WRITE(ITTY,671)ANS..6
42300      IF (ANS..6.EQ.'YES') GO TO 6666
42400      C
42500      C      CREATING INPUT MATRICE IN FILE C?????.MAT (UNIT=21)
42600      C
42700      CALL PARAM(2,NAME2)
42800      OPEN(UNIT=21,FILE=NAME2)
42900      WRITE(21,*)IZON,ITOT,K1
43000      DO 6400 I=1,IZON
43100      WRITE(21,*)XPZO(I),YPRI(I),IZOP(I),ITZO(I)
43200      6400 CONTINUE
43300      CLOSE(UNIT=21,FILE=NAME2,DISPOSE='SAVE')
43400      WRITE(ITTY,6500)NAME2
43500      6500 FORMAT(4X,'(ABOVE INFO HAS BEEN WRITTEN TO FILE ',A10,')')
43600      6666 RETURN
43700      END
43800      SUBROUTINE MATIN(XPZO,YPRI,IZON,IZOP,ITZO,ITOT)
43900      COMMENT
44000      C
44100      C      SUBROUTINE MATIN FOR READING THE MATRICE INFORMATION
44200      C
44300      COMMENT
44400      DIMENSION IZOP(21),ITZO(21),YPRI(21),XPZO(21)
44500      COMMON/COMMON1/IDUM,K1,TRAD,IPLN,ITIX,ITTY,CONT
44600      DOUBLE PRECISION NAME1
44700      WRITE(ITTY,10)
44800      10  FORMAT(/,4X,'NAME OF FILE FOR MATRICE ? ',8)
44900      READ(ITIX,20)NAME1
45000      20  FORMAT(A10)
45100      OPEN(UNIT=21,FILE=NAME1,ACCESS='SEQIN')
45200      READ(21,*)IZON,ITOT,K1
45300      DO 30 I=1,IZON
45400      READ(21,*)XPZO(I),YPRI(I),IZOP(I),ITZO(I)
45500      30  CONTINUE
45600      CLOSE(UNIT=21,FILE=NAME1,DISPOSE='SAVE')
45700      6666 RETURN
45800      END

```



```

00100 C          ***** SUB24.FOR *****
00200 C
00300 C          SUBROUTINE PLOT
00400 C
00500 C          SUBROUTINE PLOT1(IPLO,XARR,YARR,IZO1,IDEC,ZARX,ZARY,IZO2,TITL,NWO,
00600 C          1,NF0,I1I2,N#2,NF2,IEX)
00700 COMMENT
00800 C          SUBROUTINE PLOT
00900 C
01000 C          XARR AND YARR OF LENGTH IZO1 FOR POINTS
01100 C          ZARX AND ZARY OF LENGTH IZO2 FOR LINE
01200 C          IN OVERLAY (PREDICTED VALUES)
01300 C          (MAX LENGTH OF ARRAYS 250)
01400 C          IPLO 1 : NORMAL
01500 C          IPLO 2 : OVERLAY
01600 C          IDEC 1 : LINE
01700 C          IDEC 2 : POINTS
01800 C          TITL  : CHARACTER ARRAY WITH
01900 C          NWO   : NO OF WORDS OF
02000 C          NF0   : FORM OF WORDS
02100 C          IEX   : 1 ALSO VDU PLOT ; 2 NOT
02200 C
02300 C          SEE DIMENSION FOR ARRAYS
02400 COMMENT
02500 C          DIMENSION XPLO(250),YPL0(250),XARR(IZO1),YARR(IZO1),ZARX(IZO2),
02600 C          1 ZARY(IZO2),XAR1(250),YAR1(250),TITL(NWO),TIT2(NW2),
02700 C          2 ZPLX(250),ZPLY(250),TIM(3),IP(250),IST1(250)
02800 C          LOGICAL OVLA
02900 C          COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTY,CONT
03000 C          IF (IDUM.NE.1) GO TO 5555
03100 C          OVLA=.FALSE.
03200 C          IF (IPLO.EQ.2) OVLA=.TRUE.
03300 C
03400 C          SORT THE ARRAYS AFTER XARR
03500 C
03600 C          IFAI=1
03700 C          CALL M01AAF(XARR,1,IZO1,IP,IST1,IFAI)
03800 C          CALL IFTEST(IFAI,1,ITTY)
03900 C          IF (IFAI.NE.0) GO TO 6666
04000 C          DO 10 I=1,IZO1
04100 C          XAR1(IP(I))=XARR(I)
04200 C          YAR1(IP(I))=YARR(I)
04300 C          10 CONTINUE
04400 C
04500 C          FINDS MAX AND MIN
04600 C
04700 C          XMIN=0.0
04800 C          YMIN=0.0
04900 C          ZMIN=0.0
05000 C          VMIN=0.0
05100 C          XMAX=0.0
05200 C          YMAX=0.0
05300 C          ZMAX=0.0
05400 C          VMAX=0.0
05500 C          DO 4450 I=1,IZO1
05600 C          XMIN=AMIN1(XMIN,XAR1(I))
05700 C          XMAX=AMAX1(XMAX,XAR1(I))
05800 C          YMIN=AMIN1(YMIN,YAR1(I))
05900 C          YMAX=AMAX1(YMAX,YAR1(I))
06000 C          4450 CONTINUE
06100 C          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 X#ID, XHE1, 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=(X#ID-XRIO)*0.80/XDIF
08100      YTMA=AMAX1(YMAX, ZMAX)
08200      YTMI=AMIN1(YMIN, ZMIN)
08300      YDIF=YTMA-YTMI
08400      YCON=(XHE1-YRIO)*0.80/YDIF
08500      C
08600      C      CALCULATES THE NEW COORDINATES IN MM
08700      C
08800      DO 4500 I=1, IZ01
08900      XPLJ(I)=((XAR1(I)-XTMI)*XCON)+XRIO
09000      YPLJ(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 ERKMAX(50)
10500      CALL DFVPAP(X#ID, XHE1, 1)
10600      CALL MOVTO2(0.0, 0.0)
10700      CALL LINTO2(X#ID, 0.0)
10800      CALL LINTO2(X#ID, 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(XPLG(I), YPLO(I))
11800      5000 CONTINUE
11900      GO TO 5600
12000      C

```



```

12100 C IF IDEC 2 PLOTS THE POINTS OF XARR YARR
12200 C
12300 5500 CONTINUE
12400 CALL SYMTU2(XPLO,YPLO,IZO1,8)
12500 5600 CONTINUE
12600 IF (.NOT.OVLA) GO TO 5010
12700 C
12800 C PLOTS THE LINE THROUGH POINTS ZARX ZARY
12900 C
13000 CALL MOVTU2(ZPLX(1),ZPLY(1))
13100 DO 5700 I=2,IZO2
13200 CALL LINTU2(ZPLX(I),ZPLY(I))
13300 5700 CONTINUE
13400 5010 CONTINUE
13500 C
13600 C AXIS SCALE AND POSITION
13700 C
13800 CALL AXIPUS(1,XRIO,YRIO,(0.80*(XWID-XRIO)),1)
13900 CALL AXIPUS(1,XRIO,YRIO,(0.80*(XHE1-YRIO)),2)
14000 CALL AXISCA(3,10,XIMI,XIMA,1)
14100 CALL AXISCA(3,10,YIMI,YIMA,2)
14200 CALL AXIDRA(-1,1,1)
14300 CALL AXIDRA(1,-1,2)
14400 C
14500 C TEXT OUTPUT ETC.
14600 C
14700 CALL MOVTU2(70.0,160.0)
14800 CALL CHARR(TITL,NW0,NF0)
14900 CALL MOVTU2(69.0,159.0)
15000 IF (OVLA) CALL LINTU2(121.0,159.0)
15100 IF (.NOT.OVLA) CALL LINTU2(133.0,159.0)
15200 IF (.NOT.OVLA) GO TO 5020
15300 CALL MOVTU2(80.0,140.0)
15400 CALL CHARR(TIT2,NW2,NF2)
15500 CALL MOVTU2(55.0,10.0)
15600 CALL CHAHL(23HNO,OF VISITORGRUPS/*.)
15700 CALL CHAINT(K1,10)
15800 GO TO 5030
15900 5020 CONTINUE
16000 CALL MOVTU2(55.0,10.0)
16100 CALL CHARR(TIT2,NW2,NF2)
16200 5030 CONTINUE
16300 CALL MOVTU2(1.5,1.5)
16400 C CALL CHARR(TIM,2,5)
16500 CALL CHAHL(1,0)
16600 CALL CHAHL(10HPLOT NO:*. )
16700 CALL CHAINT(IPLN,-3)
16800 CALL MOVTU2(15.0,70.0)
16900 CALL CHAHL(0,1)
17000 CALL CHAHL(28HWILLINGNESS TO PAY (PENGE)*.)
17100 CALL DEVENO
17200 WRITE(ITY,5100)IPLN
17300 5100 FORMAT(4X,'( GINO-F PLOT NO :',I3,')',/)
17400 IPLN=IPLN+1
17500 IF (IEX.NE.1) GO TO 5555
17600 C
17700 C OUTPUT ON VDU
17800 C
17900 CALL ITY
18000 CALL DEVPA(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,IZO1,0)
18400 CALL DEPEND
18500 5555 CONTINUE
18600 6666 RETURN
18700 END
```



```

00100 C ***** SUB35.FOR *****
00200 C
00300 C SUBROUTINES TRNF GLM2 LSF12 HEIGHT HEADIN
00400 C FUN1 FUN2
00500 C
00600 C
00700 C
00800 C
00900 C SUBROUTINE TRNF(IWEI,XARR,YARR,INOB,XWEI)
01000 COMMENT
01100 C SUBROUTINE TRNF LN(Y)=LN(A)-B*LN(X)
01200 C XARR AND YARR OF LENGTH INOB
01300 C IWEI : 1 NORMAL TRNF
01400 C 2 - 7 WEIGHTED
01500 C
01600 C SEE DIMENSION STATEMENT FOR ARRAYS
01700 C ESTIMATED PARAMETERS IN COMMON VAR1/ XRES(20)
01800 COMMENT
01900 C DIMENSION XARR(INOB),YARR(INOB),TITL(10),
02000 C 1 TIT2(10),TIT3(10),XAR1(21),YAR1(21),XWEI(INOB),XAR2(21),
02100 C 2 YAR2(21),XPRP(100),YPRP(100)
02200 C COMMON/DUM1/IDUM,K1,TRAD,IPLN,IITX,ITTY,CONT
02300 C COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
02400 C EXTERNAL FUN2
02500 C CONT=.TRUE.
02600 C
02700 C OUTPUTS HEADING
02800 C
02900 C WRITE(ITTY,10)
03000 10 FORMAT(1H1,/,4X,'TRANSFORMED LIN. REG.:',/,4X,'*****')
03100 C 1***',/)
03200 C IF (IWEI.NE.1) CALL HEADIN(IWEI,ITTY)
03300 C CALL PRESS(3,XARR,YARR,INOB,XWEI,XPRES)
03400 C TO PREVENT EXTREMES WHEN TRANSFORMING 1.0 IS ADDED
03500 C DO 15 I=1,INOB
03600 C XAR2(I)=ALOG(XARR(I)+1.0)
03700 C YAR2(I)=ALOG(YARR(I)+1.0)
03800 15 CONTINUE
03900 C IFAIL=1
04000 C CALL SORT1(XARR,YARR,INOB,XAR1,YAR1,IFAIL,ITTY)
04100 C IF (IFAIL.NE.0) CONT=.FALSE.
04200 C IF (IFAIL.NE.0) GO TO 6666
04300 C
04400 C CALL TO SUBROUTINE GLMMY
04500 C
04600 C CALL GLMMY(INOB,XAR2,YAR2,XWEI)
04700 C
04800 C CALCULATES SS-RES FOR ORIGINAL POINTS ON BASIS OF WEIGHTED
04900 C ANALYSIS PARAMETERS
05000 C
05100 C SSRES=0.0
05200 C DO 20 I=1,INOB
05300 20 SSRES=SSRES+((YARR(I)-FUN2(XARR(I))))**2)
05400 30 CONTINUE
05500 C
05600 C POINTS FOR PREDICTED LINE
05700 C
05750 C
05800 C IF (IDUM.NE.1) GO TO 200
05850 C XDIF=XAR1(INOB)-XAR1(1)
05900 C XINI=XDIF/99

```

```

06000 35  CONTINUE
06100      DO 40 I=1,100
06200      XPRP(I)=XARR(1)+(XINT*(I-1))
06300      YPRP(I)=FUN2(XPRP(I))
06400      INO=I
06500  C
06600  C  STOPS IF YPRP NEGATIVE  (MEANS NEGATIVE VISITORS)
06700  C
06800      IF (YPRP(I).LT.0.0) GO TO 100
06900 40  CONTINUE
07000      GO TO 200
07100  C
07200  C  FINDS THE INTERSECT OF YAXIS AND CURVE
07300  C      (STOPS DRAWING NEGATIVE VISITORGROUPS)
07400 100 CONTINUE
07500      XMAX=XPRP(INO)*1.1
07600      IFAI=1
07700      EPS=0.0005
07800      ETA=0.005
07900  C
08000  C  CALLS  NAG-SUBROUTINE
08100  C
08200      CALL COSACF(0.0,XMAX,EPS,ETA,FUN2,XSTA,IFAI)
08300      CALL IFTEST(IFAI,2,ITTY)
08400      IF (IFAI.NE.0) CONT=.FALSE.
08500      IF (IFAI.NE.0) GO TO 6666
08600      XINI=(XSTA-XARR(1))/99.0
08700      GO TO 35
08800 200 CONTINUE
08900  C
09000  C  OUTPUT RESULT
09100  C
09200      WRITE(ITTY,50)XRES(8),XRES(9)
09300 50  FORMAT(4X,'F-VALUE  =',F10.5,/,4X,'R-SQUARE  =',4X,F6.5)
09400      WRITE(ITTY,65)XRES(6)
09500 65  FORMAT(4X,'SS-RES   =',E15.6)
09600      WRITE(ITTY,80)SSRES
09700 80  FORMAT(4X,'SS-RES BASED ON ORIGINAL POINTS =',E15.6)
09800      WRITE(ITTY,75)XPPES
09900 75  FORMAT(4X,'PRESS-value =',F15.2,/)
10000      WRITE(ITTY,60)XRES(1),XRES(2)
10100 60  FORMAT(4X,'REGRESSION CONSTANT:   A =',E15.5,/,4X,
10200      'REGRESSION COEFFICIENT: B =',E15.5,/)
10300      WRITE(ITTY,70)
10400 70  FORMAT(6X,'( LN(Y) = A - B*LN(X) )',/)
10500  C
10600  C  PLOTS THE TRIP DEMAND CURVE
10700  C
10800      DATA TIT2/'LINE FROM TRNF PROCEDURE*.'/
10900      DATA TIT3/'LINE FROM WEIGHTED TRNF PROCEDURE*.'/
11000      DATA TITL/'TRIP DEMAND CURVE*.'/
11100      IF (IWEI.EQ.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRP,XPRP,100,
11200      TITL,4,5,TIT2,5,5,2)
11300      IF (IWEI.NE.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRP,XPRP,100,
11400      TITL,4,5,TIT3,7,5,2)
11500 6666 RETURN
11600      END
11700      SUBROUTINE LSF12(IWEI,XARR,YARR,INOB,XWEI)
11800  COMMENT
11900  C

```



```

12000 C SUBROUTINE LSF12 2ND DG POLYNOMIAL
12100 C
12200 C SEE GLM2 FOR PARAMETERS OF CALL
12300 C EST. PARAMETERS STORED IN COMMON VAR1/ XRES2
12400 C
12500 COMMENT
12600 DIMENSION XARR(INOB),YARR(INOB),XNEI(INOB),TITL(10),YAR1(21),
12700 1 XPRP(100),YPRP(100),TIT2(10),TIT3(10),YPRE(21),
12800 2 XAR1(21),XAR2(21)
12900 COMMON /VAR1/XRES(20),XRES2(20),XC2F(3)
13000 COMMON /UM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT
13100 EXTERNAL FUN1
13200 CONT=.TRUE.
13300 IFA1=1
13400 C
13500 C OUTPUTS HEADING
13600 C
13700 WRITE(ITTY,4380)
13800 4380 FORMAT(1H1,/,4X,'2ND DG POLYNOMIAL FIT:',/,4X,'*****
13900 1+***',/)
14000 IF (IWEI.NE.1) CALL HEADIN(IWEI,ITTY)
14100 CALL PRESS(2,XARR,YARR,INOB,XNEI,XPRES)
14200 CALL SORT1(XARR,YARR,INOB,XAR1,YAR1,IFA1,ITTY)
14300 IF (IFA1.NE.0) CONT=.FALSE.
14400 IF (IFA1.NE.0) GO TO 6666
14500 DO 400 I=1,INOB
14600 400 XAR2(I)=(XARR(I)**2)
14700 C
14800 C CALL TO GLMMY2
14900 C
15000 CALL GLMMY2(INOB,XARR,XAR2,YARR,XNEI)
15100 C
15200 C CALCULATES PREDICTED VALUES
15300 C
15350 IF (IDUM.NE.1) GO TO 200
15400 XDIF=XAR1(INOB)-XAR1(1)
15500 XINT=XDIF/99
15600 K=0
15700 35 CONTINUE
15800 DO 40 I=1,100
15900 XPRP(I)=XAR1(1)+(XINT*(I-1))
16000 YPRP(I)=XRES2(1)+XRES2(2)*XPRP(I)+XRES2(3)*(XPRP(I)**2)
16100 INO=I
16200 C
16300 C STOPS IF YPRP NEGATIVE (MEANS NEGATIVE VISITORS)
16400 C
16500 IF (YPRP(I).LT.0.0) GO TO 100
16600 40 CONTINUE
16700 GO TO 200
16800 C
16900 C FINDS THE INTERSECT OF YAXIS AND CURVE
17000 C (STOPS DRAWING NEGATIVE VISITORGROUPS)
17100 100 CONTINUE
17200 XMAX=XPRP(INO)*1.1
17300 IFA1=1
17400 EPS=0.0005
17500 ETA=0.005
17600 C
17700 C CALLS MAG-SUBROUTINE
17800 C

```

```

17900 CALL COSALF(0.0,XMAX,EPS,ETA,FUN1,XSTA,IFAI)
18000 CALL IFTEST(IFAI,2,ITTY)
18100 IF (IFAI.NE.0) CONT=.FALSE.
18200 IF (IFAI.EQ.0) GO TO 6666
18300 XINI=(XSTA-XAR1(1))/99.0
18400 K=K+1
18500 IF (K.LI.50) GO TO 35
18600 WRITE(ITTY,180)
18700 180 FORMAT(1,4X,"STOP ON: K IN SUBR LSF12",/)
18800 200 CONTINUE
18900 C
19000 C CALCULATES SS-RES FOR POINTS
19100 C
19200 SSPRES=0.0
19300 DO 300 I=1,INOB
19400 YPRE(I)=FUN1(XARR(I))
19500 SSPRES=((YARR(I)-YPRE(I))**2)+SSPRES
19600 300 CONTINUE
19700 C
19800 C OUTPUTS RESULTS
19900 C
20000 WRITE(ITTY,75)XRES2(6)
20100 75 FORMAT(4X,"SS-RES =",E15.6)
20200 IF (IWEI.NE.1) WRITE(ITTY,80)SSPRES
20300 80 FORMAT(4X,"SS-RES BASED ON ORIGINAL POINTS =",E15.6)
20400 WRITE(ITTY,90)XPRES
20500 90 FORMAT(4X,"PRESS-value =",F15.2,/)
20600 WRITE(ITTY,60)XRES2(1),XRES2(2),XRES2(3)
20700 60 FORMAT(4X,"COEFFICIENTS ARE :",3X,"COEF(1) ",E13.5,/,25X,
20800 1"COEF(2) ",E13.5,/,25X,"COEF(3) ",E13.5,/,
20900 2 /,6X,"( Y = COEF(1) + COEF(2)*X + COEF(3)*X^2 )",/)
21000 C
21100 C PLOTS THE TRIP DEMAND CURVE
21200 C
21300 DATA TITL/"TRIP DEMAND CURVE*."/
21400 DATA TIT2/"2ND DG POLYNOMIAL CURVEFIT*."/
21500 DATA TIT3/"2ND DG POLYNOMIAL WEIGHTED CURVEFIT*."/
21600 IF (IWEI.EQ.1) CALL PLOT1(2,YAR1,XAR1,INOB,2,YPRP,XPRP,100,
21700 1 TITL,4,5,TIT2,6,5,2)
21800 IF (IWEI.NE.1) CALL PLOT1(2,YAR1,XAR1,INOB,2,YPRP,XPRP,100,
21900 1 TITL,4,5,TIT3,8,5,2)
22000 6666 RETURN
22100 END
22200 SUBROUTINE HEADIN(IWEI,ITTY)
22300 COMMENT
22400 C
22500 C HEADINGS FOR GLM2 LSF12 EXP2
22600 C
22700 COMMENT
22800 INTEGER IWEI,ITTY
22900 IF (IWEI.EQ.2) WRITE(ITTY,4200)
23000 4200 FORMAT(4X,"( WEIGHTED ACCORDING TO NO. OF VISITORGROUPS )",/)
23100 IF (IWEI.EQ.3) WRITE(ITTY,4400)
23200 4400 FORMAT(4X,"( WEIGHTED ACCORDING TO SQRT(ZONEPOPULATION) )",/)
23300 IF (IWEI.EQ.4) WRITE(ITTY,4600)
23400 4600 FORMAT(4X,"( WEIGHTED ACCORDING TO SQRT(ZONEPOPULATION) ",/,
23500 1 20X," * NO OF VISITORGROUPS FROM ZONE )",/)
23600 IF (IWEI.EQ.5) WRITE(ITTY,4800)
23700 4800 FORMAT(4X,"( WEIGHTED ACCORDING TO ZONEPOP./VISITRATE ) ",/)
23800 IF (IWEI.EQ.6) WRITE(ITTY,5000)

```



```

3900 5000 FORMAT(4X,'( SUBJECTIVE WEIGHTING ) ',/)
4000 IF (INEI.EQ.7) WRITE(UNIT,5200)
4100 5200 FORMAT(4X,'( WEIGHTED ACCORDING TO ZONE POPULATION ) ',/)
4200 6666 RETURN
4300 END
4400 SUBROUTINE WEIGHT(INO1,XPZO,ITZO,IZOP)
4500 COMMENT
4600 C
4700 C   CREATES THE WEIGHTING ARRAYS
4800 C
4900 COMMENT
5000 INTEGER I,01,ITZO(INO1),IZOP(INO1)
5100 REAL XPZO(INO1)
5200 COMMON/WEI1/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 DG 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
7900 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,XWEI)
8900 COMMENT
9000 C   SUBROUTINE GLM2
9100 C
9200 C   XARR AND YARR OF LENGTH INOB
9300 C   INEI : 1 NORMAL GLM
9400 C           2 - 7 WEIGHTED

```

```

29500 C                               SEE DIMENSION STATEMENT FOR ARRAYS
29600 C                               ESTIMATED PARAMETERS IN COMMON VAR1/ XRES(20)
29700 C
29800 C COMMENT
29900 C     DIMENSION XARR(INOB),YARR(INOB),XPRES(2),YPRES(2),TITL(10),
30000 C     1  TIT2(10),TIT3(10),XAR1(21),YAR1(21),XWEI(INOB)
30100 C     COMMON/JUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT
30200 C     COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
30300 C     CONT=.TRUE.
30400 C
30500 C     OUTPUTS HEADING
30600 C
30700 C     WRITE(ITTY,10)
30800 C     10  FORMAT(10I,/,4X,"LINEAR REGRESSION FIT:",/,4X,"*****")
30900 C     1 10  "*****",/)
31000 C     IF (IWEI.NE.1) CALL HEADIN(IWEI,ITTY)
31100 C     CALL PRESS(1,XARR,YARR,INOB,XWEI,XPRES)
31200 C     IFAI1=1
31300 C     CALL SORT1(XARR,YARR,INOB,XAR1,YAR1,IFAI1,ITTY)
31400 C     IF (IFAI1.NE.0) CONT=.FALSE.
31500 C     IF (IFAI1.NE.0) GO TO 6666
31600 C
31700 C     CALL TO SUBROUTINE GLMMY
31800 C
31900 C     CALL GLMMY(INOB,XARR,YARR,XWEI)
32000 C     IF (INFI.EQ.1) GO TO 30
32100 C
32200 C     CALCULATES SS-RES FOR ORIGINAL POINTS ON BASIS OF WEIGHTED
32300 C     ANALYSIS PARAMETERS
32400 C
32500 C     SSRES=0.0
32600 C     DO 20 I=1,INOB
32700 C     20  SSRES=SSRES+((YARR(I)-(XRES(1)+XPRES(2)*XARR(I)))**2)
32800 C     30  CONTINUE
32900 C
33000 C     POINTS FOR PREDICTED LINE
33100 C
33150 C     IF (IDUM.NE.1) GO TO 200
33200 C     XPRES(1)=0.0
33300 C     YPRES(2)=0.0
33400 C     YPRE(1)=XRES(1)
33500 C     XPRE(2)=-XPRES(1)/XRES(2)
33600 C
33700 C     OUTPUT RESULT
33800 C
33900 C     200 WRITE(ITTY,50)XRES(8),XRES(9)
34000 C     50  FORMAT(4X,"F-VALUE  =",F10.5,/,4X,"R-SQUARE  =",4X,F6.5)
34100 C     WRITE(ITTY,75)XRES(6)
34200 C     75  FORMAT(4X,"SS-RES   =",E15.6)
34300 C     IF (IWEI.NE.1) WRITE(ITTY,80)SSRES
34400 C     80  FORMAT(4X,"SS-RES BASED ON ORIGINAL POINTS =",E15.6)
34500 C     WRITE(ITTY,90)XPRES
34600 C     90  FORMAT(4X,"PRESS-value =",F15.2,/)
34700 C     WRITE(ITTY,60)XRES(1),XRES(2)
34800 C     60  FORMAT(4X,"REGRESSION CONSTANT:  A =",E15.5,/,4X,
34900 C     1  "REGRESSION COEFFICIENT: B =",E15.5,/)
35000 C     WRITE(ITTY,70)
35100 C     70  FORMAT(6X,"( Y = A + B*X )",/)
35200 C
35300 C     PLOTS THE TRIP DEMAND CURVE

```



```
35400 C
35500 DATA IIT2/'LINE FROM GLM PROCEDURE*.'/
35600 DATA IIT3/'LINE FROM WEIGHTED GLM PROCEDURE*.'/
35700 DATA IITL/'TIP DEMAND CURVE*.'/
35800 IF (IIEI.EQ.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRE,XPRE,2,
35900 1 IITL,4,5,IIT2,5,5,2)
36000 IF (IIEI.EQ.1) CALL PLOT1(2,YARR,XARR,INOB,2,YPRE,XPRE,2,
36100 1 IITL,4,5,IIT3,7,5,2)
36200 5606 RETURN
36300 END
```

```

00100 C ***** SUB45.FOR *****
00200 C
00300 C SUBROUTINES REDUC MAXMI AGGR2 SORT1 ISTEST
00400 C
00500 C
00600 C SUBROUTINE REDUC(ITEK,XARR,YARR,IZON,IZOP,ITZO,ITOT)
00700 COMMENT
00800 C
00900 C SUBROUTINE REDUCES XARR,YARR FOR ZERO ELEMENTS
01000 C
01100 C XARR AND YARR OF LENGTH IZON
01200 C IZOP : ZONEPOPULATION
01300 C ITZO : NO OF VISITORGR. FROM ZONE
01400 C (IF ITZO=0 OBSERVATION IS EXCLUDED)
01500 C ITEK : 1 TEXT ; 2 NOT
01600 C ITTY : OUTPUT CHANNEL
01700 C
01800 C SEE DIMENSION STATEMENT FOR ARRAYS
01900 C
02000 COMMENT
02100 DIMENSION XARR(IZON),YARR(IZON),IZOP(IZON),XAR1(21),YAR1(21),IZO1(
02200 121),IIZO(IZON),ITZ1(21)
02300 COMMON/COMMON/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT
02400 K=0
02500 DO 100 I=1,IZON
02600 IF (IIZO(I).EQ.0) GO TO 50
02700 K=K+1
02800 XAR1(K)=XARR(I)
02900 YAR1(K)=YARR(I)
03000 IZO1(K)=IZOP(I)
03100 ITZ1(K)=IIZO(I)
03200 50 CONTINUE
03300 100 CONTINUE
03400 IOUT=IZON-K
03500 IZOV=K
03600 ITOT=0
03700 DO 200 I=1,IZON
03800 XARR(I)=XAR1(I)
03900 YARR(I)=YAR1(I)
04000 IZOP(I)=IZO1(I)
04100 ITZO(I)=ITZ1(I)
04200 ITOT=ITOT+IIZO(I)
04300 200 CONTINUE
04400 IF ((IOUT.EQ.0).OR.(ITEK.NE.1)) GO TO 300
04500 WRITE(ITTY,220)
04600 220 FORMAT(/,4X,'***** WARNING *****',/)
04700 WRITE(ITTY,250)IOUT
04800 250 FORMAT(/,4X,I4,' OBSERVATION(S) EXCLUDED DUE TO ZERO VISITORS',/
04900 1)
05000 300 CONTINUE
05100 RETURN
05200 END
05300 SUBROUTINE AGGR2(ITYP,IAN1,XARR,YARR,IZON,IZOP,
05400 1 ITZO,ITOT)
05500 COMMENT
05600 C
05700 C SUBROUTINE CALCULATING AGGREGATED DEMAND FROM TRIP DEMAND
05800 C
05900 C ITYP : 1 BASED ON ZONEPOINTS
06000 C : 2 BASED ON INDIVIDUALS !!!!

```

```

06100 C           : 3 REGULATES TO ACTUAL OBSERVED
06200 C           XARR AND YARR OF LENGTH IZON
06300 C           IARR1 : 1 GLM ; 2 POLYNOMIAL ; 3 TRNF
06400 C           IZOP : ZONLPOPULATION
06500 C           ITZO : NO. OF VISITORGROUJS FROM ZONE
06600 C           ITOT : TOTAL NO OF OBSERVATIONS INCLUDED IN AN.
06700 C           ITIX : INPUT CHANNEL
06800 C           ITTY : OUTPUT CHANNEL
06900 C
07000 C           SEE DIMENSION STATEMENT FOR ARRAYS
07100 C
07200 C           PARAMETERS ARE STORED IN COMMON VAR1/
07300 C
07400 COMMENT
07500 DIMENSION XARR(IZON),YARR(IZON),IZOP(IZON),YPRE(21),XNHUB(200),
07600 XAGG(250),YAGG(250),TITL(10),IIT2(10),ITZO(IZON)
07700 EXTERNAL FUN1,FUN2
07800 COMMON/VAR1/XRES(20),XRES2(20),XC2F(3)
07900 COMMON/DUM1/IDUM,A1,TRAD,IPLN,ITIX,ITTY,CONT
08000 IF (.NOT.CONT) GOTO 6066
08100 C
08200 C           INITIALIZE ARRAYS
08300 C
08400 DO 10 I=1,250
08500 XAGG(I)=0.0
08600 YAGG(I)=0.0
08700 10 CONTINUE
08800 DO 15 J=1,200
08900 XNHUB(J)=0.0
09000 15 CONTINUE
09100 CALL MAXMI(XARR,YARR,IZON,XMAX,YMAX,XMIN,YMIN)
09200 ISIA=0
09300 YINC=5.0
09400 EPS=0.0005
09500 ETA=0.005
09600 YINI=(YMAX-YMIN)/10
09700 K=0
09800 C
09900 C           FINDS YMAX FOR INTEGRATION (INTERSECT)
10000 C
10100 25 CONTINUE
10200 IFAI=1
10300 K=K+1
10400 GOTO (50,30,40)IARR1
10500 50 YMAX2=YINI+(YINI*K)
10600 CALL COSACF(0.0,YMAX2,EPS,ETA,FUN1,YSIA,IFAI)
10700 IF ((IFAI.NE.0).AND.(K.LT.25)) GO TO 25
10800 CALL IFTEST(IFA1,4,ITTY)
10900 XCM=-XRES2(2)/(XRES2(3)*2)
11000 IF ((FUN1(XCM).GT.0.0).AND.(XRES2(3).GE.0.0)) GO TO 35
11100 GO TO 60
11200 35 CONTINUE
11300 C           SUFF OPTION FOR POLYNOMIAL - INTEGRATES TO YMAX
11400 WRITE(ITTY,36)XCM
11500 36 FORMAT(/,4X,'***** WARNING *****',/,4X,'NO INTERSECT - WILL IN
11600 TEGRATE TO VALUE FOR WHICH',/,4X,'FUNCTION HAS MINIMUM - EQUAL TO'
11700 2,F8.2,' PERCENT ',/)
11800 YSIA=XCM
11900 GO TO 60
12000 40 YSIA= EXP(-XRES(1)/XRES(2))

```



```

12100      GJ 10 6J
12200      50  YSTA=-XRES(1)/XRES(2)
12300      57  YLMA=YSTA
12400      C
12500      C  TEST OF SIZE OF Y INCREMENT
12600      C
12700      CALL ISTEET(YLMA,YINC,IITY)
12800      IF (IITY.EQ.2) REWIND 22
12900      IF (IITY.EQ.2) ICONF=IZON
13000      IF (IITY.EQ.2) ICONF=ITOT
13100      DO 100 I=1,ICONF
13200      IF (IITY.EQ.2) READ(22,46) YVAR
13300      46  FORMAT(F16.2)
13400      IF (IITY.NE.2) YVAR=YARR(1)
13500      ISTEP=INT((YLMA-YVAR)/YINC)+1
13600      IF (YLMA.LE.YVAR) GO TO 100
13700      ISTM=MAX(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 (IITY.EQ.1) XNUB(J)=(XSTA*IZOP(I)/K1)+XNUB(J)
14900      IF (IITY.EQ.2) XNUB(J)=(XSTA/XPAR)+XNUB(J)
15000      IF (IITY.EQ.3) XNUB(J)=(XSTA*ITZO(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 TIT1/'AGGREGATED DEMAND CURVE*.'/
16100      DATA TIT2/'NO. OF VISITORGROUPS*.'/
16200      CALL PLOT1(1,XAGG,YAGG,ISTM,1,XAGG,YAGG,ISTM,TIT1,6,5,TIT2,5,5,1)
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      GO TO(700,1000,1100)IITY
17700      900  WRITE(IITY,800)ITOT
17800      800  FORMAT(/,4X,'ACTUAL OBSERVED NO OF VISITORGROUPS',I5)
17900      GO TO 5555
18000      1000 WRITE(IITY,754)

```

```

18100 754 FORMAT(4X, '( BASED ON INDIVIDUAL INTEGRATION )',/)
18200      GO TO 5555
18300 1100 EFFICIENCY,752)
18400 752 FORMAT(4X, '( REGULATED TO OBSERVED NO OF VISITORGROUPS )',/)
18500 5555 EFFICIENCY,802)XAGG(1)
18600 802 FORMAT(4X, 'PREDICTED FROM THE MODEL',F8.0,/)
18700 6666 RETURN
18800      END
18900      SUBROUTINE SORT1(XARR,YARR,INOB,XAR4,YAR4,IFAI,ITTY)
19000 COMMENT
19100 C
19200 C      SUBROUTINE SORT1 (XARR=ARRAY TO BE SORTED AFTER)
19300 C                      XAR4 AND YAR4 ARE THE SORTED OUTPUT ARRAYS
19400 C
19500 COMMENT
19600      DIMENSION XARR(INOB),YARR(INOB),XAR4(INOB),YAR4(INOB),IP(4000),
19700      1-IST1(4000)
19800      IFAI=1
19900      CALL MGTAAF(XARR,1,INOB,IP,IST1,IFAI)
20000      CALL IFIFST(IFAI,1,ITTY)
20100      IF (IFAI.NE.0) GO TO 6666
20200      DO 10 I=1,INOB
20300      YAR4(IP(I))=YARR(I)
20400      XAR4(IP(I))=XARR(I)
20500 10 CONTINUE
20600 6666 RETURN
20700      END
20800      SUBROUTINE ISTEST(YLMA,YINC,ITTY)
20900 COMMENT
21000 C
21100 C      SUBROUTINE CHECKING NO OF STEPS IN INTEGRATION
21200 C
21300 C                      YLMA : YMAX FOR INTEGRATION
21400 C                      (TRIP DEMAND CURVES INTERSECT WITH
21500 C                      Y-AXIS )
21600 C                      YINC : INCREMENT FOR INTEGRATION
21700 C                      ITTY : OUTPUT CHANNEL
21800 C
21900 COMMENT
22000      ISTP=INT(YLMA/YINC)
22100      IF (ISTP.LT.200) GO TO 500
22200      YINC2=YLMA/180.0
22300      WRITE(ITTY,100)YINC,YINC2
22400 100 FORMAT(/,4X,'INCREMENT IN Y FOR CALCULATING AGGREGATED DEMAND',
22500 1 /,4X,'HAS BEEN CHANGED FROM ',10X,F3.0,' TO ',F3.0,' PENCE',/)
22600      YINC=YINC2
22700 500 CONTINUE
22800      RETURN
22900      END
23000      SUBROUTINE RANNAM(ITYP,FNAME)
23100 COMMENT
23200 C
23300 C      GENERATING PSEUDOWARE FOR TMP-FILES
23400 C
23500 COMMENT
23600      DOUBLE PRECISION FNAME
23700      INTEGER IR(5),ITYP
23800      INTEGER G05DYF
23900 C
24000 C      SETS THE RANDOM GENERATOR TO A NONREPEATABLE STATE

```

```

24100 C
24200 CALL G05CCF
24300 DO 10 I=1,5
24400 C
24500 C GIVES INTEGER RANDOM NUMBERS IN THE INTERVAL 1,9
24600 C
24700 10 IR(I)=G05DYF(1,9)
24800 C
24900 C ENCODE NOT STANDARD FORTRAN
25000 C
25100 IF (IIYP.EQ.1) ENCODE(10,100,FNAME)IR
25200 IF (IIYP.EQ.2) ENCODE(10,200,FNAME)IR
25300 100 FORMAT(1HL,5I1,4H.TMP)
25400 200 FORMAT(1HL,5I1,4H.MAT)
25500 6666 RETURN
25600 END
25700 SUBROUTINE MAXMI(XARR,YARR,IZON,XMAX,YMAX,XMIN,YMIN)
25800 COMMENT
25900 C
26000 C SUBROUTINE MAXMI TO FIND MAX AND MIN OF THE TWO ARRAYS
26100 C XARR AND YARR OF LENGTH IZON
26200 C
26300 COMMENT
26400 DIMENSION XARR(IZON),YARR(IZON)
26500 XMIN=0.0
26600 YMIN=0.0
26700 XMAX=0.0
26800 YMAX=0.0
26900 DO 100 I=1,IZON
27000 XMAX=AMAX1(XMAX,XARR(I))
27100 YMAX=AMAX1(YMAX,YARR(I))
27200 XMIN=AMIN1(XMIN,XARR(I))
27300 YMIN=AMIN1(YMIN,YARR(I))
27400 100 CONTINUE
27500 RETURN
27600 END

```



```

00100 C ***** SUB55.FOR *****
00200 C
00300 C SUBROUTINES IFTEST SUBWEI GLMMY GLMMY2 PRESS
00400 C
00500 C

```

```

00600 SUBROUTINE IFTEST(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,"PROBABLY NO INTERSECT BETWEEN TRIP DEMAND CURVE ",
02500 2 " AND Y-AXIS !!",/)
02600 400 CONTINUE
02700 6666 RETURN
02800 END

```

```

02900 SUBROUTINE SUBWEI(INO1,XWEI,SURW)

```

```

03000 COMMENT

```

```

03100 C
03200 C
03300 C SUBJECTIVE WEIGHTED POINTS
03400 C

```

```

03500 COMMENT

```

```

03600 DIMENSION XWEI(INO1)
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) ? ",$)
04500 READ(ITTX,50,ERR=20)ANSW
04600 50 FORMAT(A3)
04700 IF (ANSW.NE."YES") GO TO 100
04800 WRITE(ITTY,60)
04900 60 FORMAT(/,4X,"WEIGHTING USED:",/,4X,"*****",/)
05000 DO 80 I=1,INO1
05100 WRITE(ITTY,70)I,XWEI(I)
05200 70 FORMAT(6X,"ZONE NO. ",I2," ; WEIGHT ",E15.8)
05300 80 CONTINUE
05400 GO TO 5555
05500 100 DO 200 I=1,INO1
05600 110 GO TO 130
05700 120 SKIP RECORD ITTX
05800 130 WRITE(ITTY,140)I
05900 140 FORMAT(/,6X,"ZONE NO. ",I2," WEIGHT ? ",$)
06000 READ(ITTX,*,ERR=120)XWEI(I)

```

```

06100      IF (BATCH) WRITE(ITTY,150)XWEI(I)
06200      150  FORMAT(E15.8)
06300      XTGT=XTGT+XWEI(I)
06400      200  CONTINUE
06500      5555 SUBR=.TRUE.
06600      6666 RETURN
06700      END
06800      SUBROUTINE GLMNY(IOBS,XAR,YAR,WEI)
06900  COMMENT
07000  C
07100  C      WEIGHTED GLM PROCEDURE
07200  C
07300  C      XAR YAR ARRAYS 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,SYX,SXY,ACONS,BCOEF,DF,
07900      2  SSTOT,SSRES,SSREG,MSREG,MSRES,FVAL,RSQ,XMEAN,YMEAN,STRES
08000      COMMON/VAR1/XRES(20),XPRES2(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      SYX=0.0
08900      SXY=0.0
09000  C
09100  C      CALCULATING SS
09200  C
09300      DO 300 I=1,IOBS
09400      DF=DF+DBLE(WEI(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      SYX=SYX+(DBLE(YAR(I))**2)*DBLE(WEI(I))
09900      SXY=SXY+DBLE(XAR(I))*DBLE(YAR(I))*DBLE(WEI(I))
10000      300  CONTINUE
10100      PCOEF=(SXY-(SX*SY/DF))/(SXX-((SX**2)/DF))
10200      ACONS=SY/DF-PCOEF*SX/DF
10300      SSTOT=SYX-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)=SMGL(MSRES)
12200      XRES(8)=SMGL(FVAL)
12300      XRES(9)=SMGL(RSQ)
12400      XRES(10)=SMGL(XMEAN)
12500      XRES(11)=SMGL(YMEAN)
12600      XRES(12)=SMGL(DF)
12700      XRES(13)=SMGL(SSTOT)
12800      6666 RETURN
12900      END
13000      SUBROUTINE GLMXY2(IJBS,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 IJBS
14000      REAL XAR1(IJBS),XAR2(IJBS),YARR(IJBS),WEI(IJBS)
14100      DOUBLE PRECISION E11,E21,E31,E12,E22,E32,E13,E23,E33,DET,
14200      1 SX1,SX2,SX1X1,SX1X2,SX2X2,SY,SYX,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      SYX=0.0
15600      SX1Y=0.0
15700      SX2Y=0.0
15800  C
15900  C      CALCULATING SS
16000  C
16100      DO 100 I=1,IJBS
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      SYX=SYX+(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 CALCULATING B1 B2 B3
20200 C
20300 B1= F11*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 CALCULATING SSTOT SSRES ETC
20800 C
20900 SSTOT=SY*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(P1)
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

```

```

24100 C          2 LSF12
24200 C          3 TRFN
24300 C          X AN, Y : ARRAYS OF LENGTH INOB
24400 C          INOB: NO OF OBSERVATIONS
24500 C          XWEI: ARRAY OF LENGTH INOB WITH WEIGHTS
24600 C          XPRES: RETURNS PRESS VALUE
24700 C
24800 COMMENT
24900          DIMENSION X(INOB),Y(INOB),XNEW(21),YNEW(21)
25000          1 ,X2NEW(21),XLOG(21),YLOG(21),XWEI(INOB)
25100          COMMON/DUM1/IDUM,K1,TRAD,IPLN,ITTX,ITTY,CONT/VAR1/XRES(20),
25200          1 XRES2(20),XC2F(3)
25300          EXTERNAL FUN2
25400          XPRES=0.0
25500          DO 1000 I=1,INOB
25600          K=0
25700          DO 500 J=1,INOB
25800          IF (J.FW.1) GO TO 500
25900          K=K+1
26000          XNEW(K)=X(J)
26100          X2NEW(K)=X(J)**2
26200          YNEW(K)=Y(J)
26300          500 CONTINUE
26400          INEW=INOB-1
26500 COMMENT FUNCTION TO TEST
26600          GOTJ (520,540,560)IAN
26700          520 CALL GLM1Y(INEW,XNEW,YNEW,XWEI)
26800          YPRE=XRES(1)+XRES(2)*X(I)
26900          GO TO 560
27000          540 CALL GLM1Y2(INEW,XNEW,X2NEW,YNEW,XWEI)
27100          YPRE=XRES2(1)+XRES2(2)*X(I)+XRES2(3)*(X(I)**2)
27200          GO TO 580
27300          560 DO 565 ICOUN=1,INEW
27400          XLOG(ICOUN)=ALOG(XNEW(ICOUN)+1.0)
27500          565 YLOG(ICOUN)=ALOG(YNEW(ICOUN)+1.0)
27600          CALL GLM1Y(INEW,XLOG,YLOG,XWEI)
27700          YPRE=FUN2(X(I))
27800          GO TO 580
27900          580 CONTINUE
28000          XADD=(Y(I)-YPRE)**2
28100          XPRES=XPRES+XADD
28200          1000 CONTINUE
28300          RETURN
28400          END

```

. Appendix A5.2

.Computer Programme PFAFO1.FOR


```

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 FNAME(7),XDIS2(100,2)
01900      COMMON/COD0/FNAME(7),IZUP(200),INOV(200),IZNO(200),COUNT(1),
02000      IFORN,ASTAY,LOG1(200),LOG2(200),CFOND
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),YCDS(200)
02600      TEST=.TRUE. ! IF TEST TRUE TESTFILE = FOR31.DAT
02700      CALL PCOIN(TEST)
02800      CALL DISIN(TEST)
02900      CALL NEARN(TEST)
03000      CALL MLOT ; 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),CFOND,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),CFOND,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),IPOPUS,S100)
07300      CALL DISTA(IFURN,IX(I),XDIST,S100)
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(IPOPUS) ! 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)=IPOPUS
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),IPOPUS,S300)
11600      IZOP(JS)=IPOPUS
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+1,1),IPOP(J+1,2),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

```



```

100 IF ((IPOCO,GE,2000).AND,(IPOCO,LT,2100)) IPOCK=2000
200 IF ((IPOCO,GE,2100).AND,(IPOCO,LT,2200)) IPOCK=2100
300 IF ((IPOCO,GE,2200).AND,(IPOCO,LT,2300)) IPOCK=2200
400 IF ((IPOCO,GE,2300).AND,(IPOCO,LT,2400)) IPOCK=2300
500 IF ((IPOCO,GE,2400).AND,(IPOCO,LT,2450)) IPOCK=2400
600 IF ((IPOCO,GE,2450).AND,(IPOCO,LT,2500)) IPOCK=2450
700 IF (IPOCO,EQ,4060) GO TO 50
800 IF ((IPOCO,GE,2500).AND,(IPOCO,LT,4100)) GO TO 100
900 IF (IPOCO,EQ,4621) GO TO 100
000 IF (IPOCO,EQ,4622) GO TO 100
100 50 RETURN ! ; RETURNS TO CALLING LINE
200 100 CALL PDPUS(IPOCD,IPOPU,S75) ! CHECK POPULATION AND DISTANCE
300 CALL DISTA(IFORN,IPOCO,XDIST,S85) ! IN INT 2500 TO 4100
400 IPOCK=IPOCO ! - IF NOT EXIST, GO TO 50 AND RETURN
500 RETURN 1 ! IPOCK=999 - OTHERWISE TO LINENUM IN CALL
600 C TO CHECK IF ALL POPULATIONS EXIST
700 75 WRITE(32,80)IPOCO
800 80 FORMAT(4X,'POPULATION DOES NOT EXIST FOR POSTZONE ',I5,' !! ')
900 GO TO 50
000 85 WRITE(32,90)IFORN,IPOCO
100 90 FORMAT(4X,'NO DISTANCE FOUND FOR FOREST NO.',I3,' TO POSTCODE NO.',
200 ',I5,' !! ')
300 GO TO 50
400 END
500 C
600 C PCAGG
700 C
800 SUBROUTINE PCAGG(J,JR)
900 DIMENSION IX2(200),IY2(200)
000 COMMON/COD2/IX(200),IY(200)
100 J0999=0;J1000=0;J1500=0;J2000=0;J2100=0;J2200=0;J2300=0;J2400=0;
200 J2450=0
300 L0999=,FALSE.;L1000=,FALSE.;L1500=,FALSE.;L2000=,FALSE.;L2100=,FAL
400 SE.;L2200=,FALSE.;L2300=,FALSE.;L2400=,FALSE.;L2450=,FALSE.
500 K=0
600 DO 5000 I=1,J
700 IF (IY(I),EQ,0) GO TO 5000
800 CALL PCCHA(IX(I),IPOCK,S5000)
900 999 IF (IPOCK,NE,999) GO TO 1000
000 J0999=J0999+IY(I)
100 L0999=,TRUE.
200 1000 IF (IPOCK,NE,1000) GO TO 1500
300 J1000=J1000+IY(I)
400 L1000=,TRUE.
500 1500 IF (IPOCK,NE,1500) GO TO 2000
600 J1500=J1500+IY(I)
700 L1500=,TRUE.
800 2000 IF (IPOCK,NE,2000) GO TO 2100
900 J2000=J2000+IY(I)
000 L2000=,TRUE.
100 2100 IF (IPOCK,NE,2100) GO TO 2200
200 J2100=J2100+IY(I)
300 L2100=,TRUE.
400 2200 IF (IPOCK,NE,2200) GO TO 2300
500 J2200=J2200+IY(I)
600 L2200=,TRUE.
700 2300 IF (IPOCK,NE,2300) GO TO 2400
800 J2300=J2300+IY(I)
900 L2300=,TRUE.
000 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=L+1
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,$100)
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)
```

6100
6200
6300
6400
6500
6600
6700
6800
6900
7000
7100
7200
7300
7400
7500
7600
7700
7800
7900
8000
8100
8200
8300
8400
8500
8600
8700
8800
8900
9000
9100
9200
9300
9400
9500
9600
9700
9800
9900
A0000
A0100
A0200
A0300
A0400
A0500
A0600
A0700
A0800
A0900
A1000
A1100
A1200
A1300
A1400
A1500
A1600
A1700
A1800
A1900
A2000

```

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
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
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
C   SUBROUTINE MPLOT
COMMON/COD0/FNAME(7),IZOP(200),INDV(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
C   ENTRY PLOT2(INU)
XMIN=0.10

```



```

4 2100 XMAX=0.40
4 2200 YMIN=0.10
4 2300 YMAX=0.40
4 2400 IPLO=INU-INT(INU/4,0)*4,0
4 2500 IF (IPLO,EQ,1) YFAC=0,5
4 2600 IF (IPLO,EQ,1) XFAC=0,0
4 2700 IF (IPLO,EQ,2) XFAC=0,0
4 2800 IF (IPLO,EQ,2) YFAC=0,0
4 2900 IF (IPLO,EQ,3) YFAC=0,5
4 3000 IF (IPLO,EQ,3) XFAC=0,5
4 3100 IF (IPLO,EQ,0) XFAC=0,5
4 3200 IF (IPLO,EQ,0) YFAC=0,0
4 3300 CALL PSPACE(XMIN+XFAC,XMAX+XFAC,YMIN+YFAC,YMAX+YFAC)
4 3400 CALL MAXMI(XPCT,YCOS,JS,XMIN2,XMAX2,YMIN2,YMAX2)
4 3500 C IF ONLY POINTS FOR WHICH DEMAND > 0 WANTS PLOTTED == CHANGE TO JORI
4 3600 C IN LINE 43300 , 43900 AND 46600
4 3700 CALL MAP(XMIN2,XMAX2*1.1,YMIN2,YMAX2*1.1)
4 3800 CALL CTRMAG(15)
4 3900 CALL PTPLOT(XPCT,YCOS,1,JS,45)
4 4000 C DO 200 I=1,JS
4 4100 C200 CALL POINT(XPCT(I),YCOS(I))
4 4200 CALL BORDER
4 4300 J=0
4 4400 CALL CTRMAG(8)
4 4500 IF (XMAX2.GT,1,0) GO TO 250
4 4600 5 J=J+1 ! FOR CHECK ON X-AXIS
4 4700 IF ((10,0**J)*XMAX2.LT,1,0) GO TO 5
4 4800 XMAX3=XMAX2*(10,0**J)
4 4900 XINT3=FLOAT(INT(XMAX3+1,0))/10,0
4 5000 CALL MAP(XMIN2,XMAX3*1.1,YMIN2,YMAX2*1.1)
4 5100 CALL AXESSI(XINT3,0)
4 5200 GO TO 275
4 5300 250 CALL AXESS
4 5400 275 CALL CTRMAG(-1)
4 5500 CALL PSPACE(0,0+XFAC,0,5+XFAC,0,0+YFAC,0,5+YFAC)
4 5600 CALL MAP(0,0,100,0,0,0,100,0)
4 5700 CALL CTRMAG(15)
4 5800 CALL PLOTCS(15,0,90,0,FNAME,35)
4 5900 C CALL CTRMAG(-1)
4 6000 CALL CTRMAG(10)
4 6100 CALL PLOTCS(45,0,10,0,'VISIT RATE',10)
4 6200 CALL POSITN(22,0,83,0)
4 6300 CALL TYPECS(COUNT,5)
4 6400 CALL SPACE(1)
4 6500 CALL TYPENI(CFOND)
4 6600 CALL SPACE(5)
4 6700 CALL TYPECS('ZONES :',7)
4 6800 CALL SPACE(1)
4 6900 CALL TYPENI(JS)
4 7000 IF (J,EQ,0) GO TO 300
4 7100 CALL POSITN(75,0,10,0)
4 7200 CALL TYPENI(10)
4 7300 CALL SUPFIX
4 7400 CALL TYPENI(-J)
4 7500 CALL NORMAL
4 7600 300 CALL CTRORI(1,0)
4 7700 CALL PLOTCS(6,0,40,0,'TRAVEL COST (KM)',16)
4 7800 CALL CTRORI(0,0)
4 7900 C CALL CTRMAG(-1)
4 8000 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 ? ',S)
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 = ',S)
02900 READ(ITTX,63)ALO1
03000 63 FORMAT(A1)
03100 WRITE(ITTX,64)
03200 64 FORMAT(4X,'EXCLUDE OBS IF LOG2 = ',S)
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.ALD1) GO TO 50
04300 IF (ALG2(I).EQ.ALD2) 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(ITY,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 ? ',S)
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 ) ? ',S)
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 = ',S)
04700 READ(ITTX,*)XA1
04800 WRITE(ITTX,76)
04900 76 FORMAT(8X,'B1 = ',S)
05000 READ(ITTX,*)XB1
05100 WRITE(ITTX,77)
05200 77 FORMAT(/,4X,'PARAMETERS FROM 2nd STAGE EXPFIT:',//,
05300 8X,'A2 = ',S)
05400 READ(ITTX,*)XA2
05500 WRITE(ITTX,78)
05600 78 FORMAT(8X,'B2 = ',S)
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(ITY,80)XA1,XB1
06200 80 FORMAT(//,4X,'PARAMETERS FROM 1st STAGE EXPFIT:',//,
06300 . 8X,'A1 = ',F12.8,/,8X,'A2 = ',F12.8,/)
06400 WRITE(ITY,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 = ',S)
06900 READ(ITTX,63)ALO1
07000 63 FORMAT(A1)
07100 WRITE(ITTX,64)
07200 64 FORMAT(4X,'EXCLUDE OBS IF LOG2 = ',S)
07300 READ(ITTX,63)ALO2
07400 WRITE(ITY,66)ALO1,ALO2
07500 66 FORMAT(/,4X,'EXCLUDE IF LOG1 = ',A1,/,
07600 . ,4X,'EXCLUDE IF LOG2 = ',A1,/)
07700 JCDU=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 JCDU=JCDU+1
08500 IX(JCDU)=IX(I)
08600 IY(JCDU)=IY(I)
08700 IZOP(JCDU)=IZOP(I)
08800 XPCT(JCDU)=XPCT(I)
08900 YCOS(JCDU)=YCOS(I)
09000 ALG1(JCDU)=ALG1(I)
09100 ALG2(JCDU)=ALG2(I)
09200 ITOT=ITOT+IY(JCDU)
09300 50 CONTINUE
09400 WRITE(ITY,250)JCDU,JR
09500 250 FORMAT(/,4X,I3,' OBSERVATIONS OUT OF ',I3,' SELECTED',/)
09600 JR=JCDU
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 ; XNTU=0.0
10300 WRITE(ITY,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 XNTU=XNTU+XNO(I) ! SUMS NO OF VISITORGROUPS
11000 XTOT=XTOT+XCS(I) ! SUMS CONSUMERS SURPLUS
11100 ITMP=INT(XNO(I))
11200 500 WRITE(ITY,505)IX(I),IY(I),ITMP,XCS(I)
11300 505 FORMAT(4X,3(4X,I6),4X,F12.0)
11400 INNO=INT(XNTU+0.5)
11500 WRITE(ITY,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) ? ',S)

```



```

12100      READ(ITTX,74)ANSW2
12200      IF (ANSW2,EQ,'YES') GO TO 3
12300      GO TO 666
12400      6666 WRITE(ITYY,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),YCJS(200)
16300      COMMON/COD6/XA1,XB1,XA2,XB2,ALO1,ALO2
16400      DOUBLE PRECISION NAM
16500      EXTERNAL MYFN1,MYFN2,MYFN3
16600      ITTX=5;ITYY=28
16700      WRITE(ITTX,10)
16800      10  FORMAT(/,4X,'NAME FOR GRIDFILE ? ',S)
16900      READ(ITTX,15)NAM
17000      15  FORMAT(A10)
17100      WRITE(ITYY,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 PTPLT(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(CFONO)
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 SUPFIX
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 SUPFIX
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 SUPFIX
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
0000 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          ALU11=ALU1

```

```

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 SUPFIX
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

VANE OF FOREST : HARESKOVENE

01K9H 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	380	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
3090	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 LANES AGRICULTURAL TRUST (ROTHAMSTED EXPERIMENTAL STATION)

- 1 'REFERENCE' EXPW01
- 2 'PROGRAM GENSTAT 00 1981-09-16 ''
- 3 'STFP 1 EXPFIT TO ESTIMATE VARIANS ''
- 4 'SCALAR' ILEN
- 5 'HEAD' HEA2
- 6 'HEAD' HLO1
- 7 'HEAD' HLO2
- 8 'INPUT' 2
- 9 'READ/S' HEA2
- 10 'READ/S' HLO1
- 11 'READ/S' HLO2
- 12 'READ' ILEN
- 13 'INPUT' 1
- 14 'PRINT' HEA2
- 15 'PRINT' HLO1,HLO2
- 16 'PRINT' ILEN
- 17 'RUN'

HARESKOVENE OG JONSTRUP VANG

A
A

ILEN 7.0000E 1

- 18 'UNIT' MYOB \$ ILEN
- 19 'VARIATE' COST \$ ILEN
- 20 'VARIATE' VISI \$ ILEN


```

21 *VARIATE* IZOP & ILEN
22 *VARIATE* RES1 & ILEN
23 *NAMES* VARL=T,F
24 *FACTOR* LOG1 & VARL,ILEN
25 *FACTOR* LOG2 & VARL,ILEN
26 *INPUT* 2
27 *READ/P* COST,VISI,IZOP,LOG1,LOG2
28 *INPUT* 1
29 *PRINT* HEA2
30 *PRINT/P* COST,VISI,IZOP,LOG1,LOG2
31 *MODEL* MYFUN & MR=A*EXP(B*COST)
32 *SCALAR* A=5.5 : B=-0.070
33 *VARIATE* STEPL= 0.01 , 0.005
34 *OPTIMIZE / LIK= 2 , NPAR= 2 , PRINT=MPF , ITER=50 * MODEL= MYFUN
35 ; PARAM=A,3;STEPS=STEPL;Y=VISI;Z=MR ; RES = RES1
36 *OUTPUT* 2
37 *PRINT/P* A,R & 12.8 , 12.8
38 *OUTPUT* 1
39 *RUN*

```

COST	MNMI VMAX	37.7514	5.4000	69.6000	70	VALUES
VISI	MNMI VMAX	0.0926	0.0000	1.6283	70	VALUES
IZOP	MNMI VMAX	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	1.3880E	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	1.7108E	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 ***

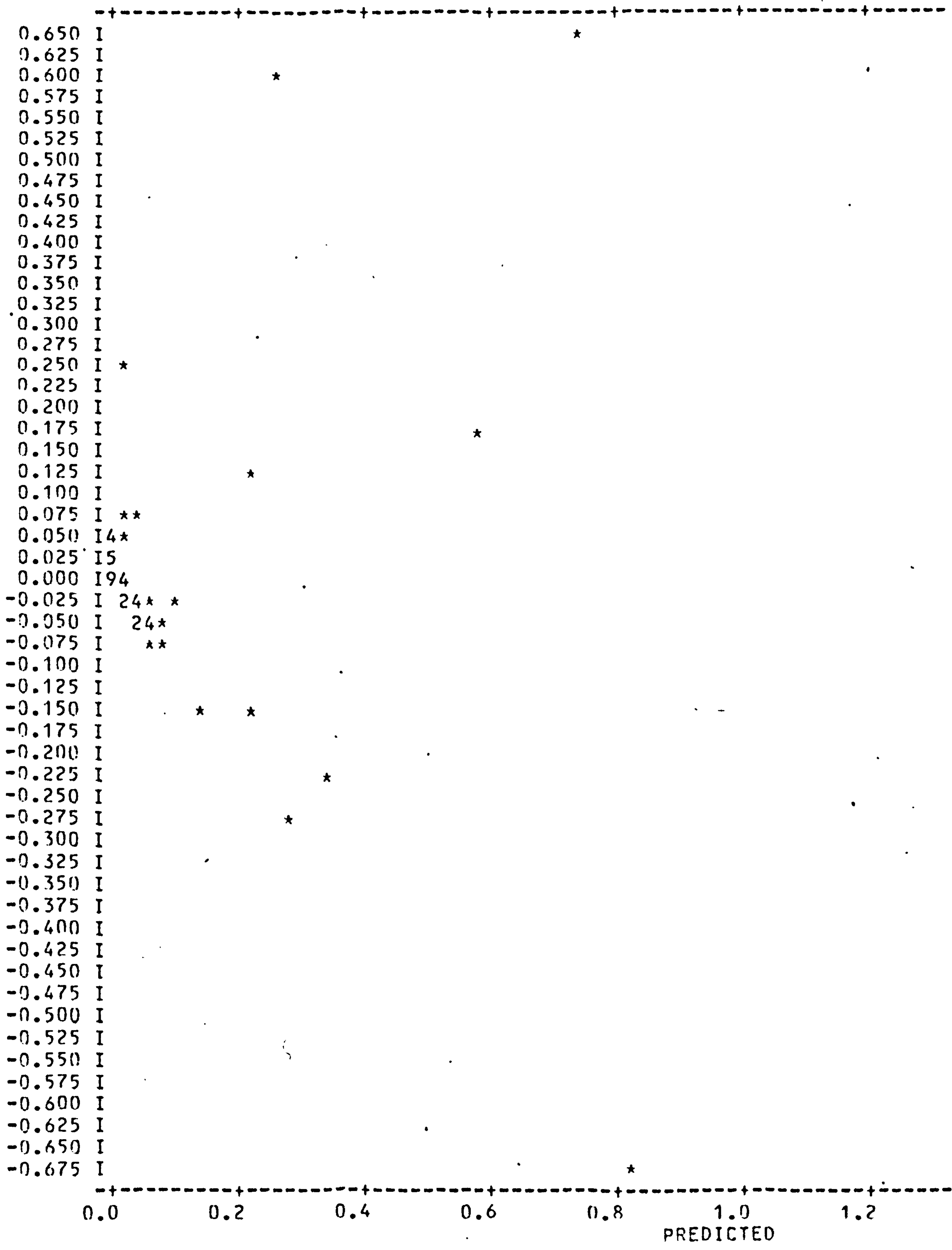
UNIT	OBSERVED	FITTED	RESIDUAL
1	0.0132	0.0083	0.0049
2	0.0127	0.0132	-0.0005
3	0.0319	0.0212	0.0107
4	0.0317	0.0072	0.0244
5	0.0090	0.0664	-0.0575
6	0.0029	0.0052	-0.0023
7	0.0244	0.0185	0.0058
8	0.0266	0.0075	0.0191
9	0.0026	0.0116	-0.0090
10	0.1012	0.0192	0.0820
11	0.0123	0.0388	-0.0265
12	0.2608	0.0198	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

68



RESIDUAL

```

46  ** STEP 2 USING VARIANCES FROM STEP 1 **
47  'VARIATE' MR2 $ ILEN
48  'CALC' MP2=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,WT=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	1.3380E	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	3.9432E	3
1.4600F	1	1.1959E	-1	5.8200E	3	F	1.6397E	4
3.5800E	1	6.5975E	-2	3.7590E	3	F	3.8392E	5
2.7400E	1	2.1853E	-2	2.2880E	3	F	5.7009E	4
6.9600E	1	5.3794E	-2	1.0410E	3	F	3.1041E	7
6.6400E	1	0.0000E	0	3.5780E	3	F	6.2334E	7
5.6600E	1	0.0000E	0	2.3620E	3	F	7.9354E	6
6.1600E	1	0.0000E	0	5.7310E	3	F	4.4588E	7
2.4600E	1	0.0000E	0	8.4400E	3	F	1.3140E	5
2.6000E	1	0.0000E	0	6.0550E	3	F	1.1926E	5
2.9000E	1	0.0000E	0	3.1250E	3	F	1.0187E	5
2.6800E	1	0.0000E	0	1.0634E	4	F	2.3956E	5
2.6800E	1	0.0000E	0	2.0880E	3	F	4.7039E	4
3.4000E	1	0.0000E	0	8.0400E	2	F	6.0693E	4
3.6400E	1	0.0000E	0	2.6970E	3	F	3.0466E	5
4.9400E	1	0.0000E	0	2.1710E	3	F	2.1767E	6
4.3800E	1	0.0000E	0	6.4190E	3	F	2.5127E	6
4.8000E	1	0.0000E	0	3.5490E	3	F	2.8127E	6
5.7800E	1	0.0000E	0	1.5670E	3	F	6.4400E	6
6.1800E	1	0.0000E	0	3.3820E	3	F	2.7211E	7
6.4400E	1	0.0000E	0	3.8690E	3	F	4.8174E	7
6.9000E	1	0.0000E	0	4.4400E	2	F	1.1970E	7
5.3400E	1	0.0000E	0	1.1760E	3	F	2.3083E	6
4.4400E	1	0.0000E	0	3.7000E	2	F	1.6019E	5
3.6800E	1	0.0000E	0	2.3200E	2	F	2.8028E	4
4.5200E	1	0.0000E	0	1.1545E	4	F	5.7172E	6
2.8400E	1	0.0000E	0	5.6000E	3	F	1.6505E	5
2.5800E	1	0.0000E	0	7.7510E	3	F	1.4762E	5
2.5200E	1	0.0000E	0	4.5000E	1	F	7.7488E	2
6.3200E	1	0.0000E	0	3.9370E	3	F	4.0073E	7
6.4400E	1	0.0000E	0	5.3000E	2	F	6.5991E	6
2.0000E	1	0.0000E	0	3.5000E	1	F	2.5166E	2
3.0200E	1	0.0000E	0	1.4070E	3	F	5.6106E	4
4.2800E	1	0.0000E	0	2.4420E	3	F	8.0814E	5
5.7800E	1	0.0000E	0	5.5440E	3	F	2.2785E	7
6.9600E	1	0.0000E	0	2.3060E	3	F	6.8762E	7
2.3800E	1	0.0000E	0	7.3300E	2	F	9.9773E	3
6.8400E	1	0.0000E	0	6.3100E	2	F	1.5381E	7
6.3800E	1	0.0000E	0	2.7051E	4	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	0.0095	-0.0197	0.0583
6	0.0029	0.0137	-0.0103	0.9078
7	0.0244	0.0353	-0.0039	0.1288
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	0.0887	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	0.0883	-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 I 7** * *
 -0.0025 I 22*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

RESIDUAL

74

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	88	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