1/9

UR1983_26

1983/26. FORTRAN programs for verifying the areal distribution of data points (Revision 1).

R.G. Richardson

Abstract

These programs check for duplicated sample point numbers, sample points closer together than a specified distance, and no sample taken within a specified distance of a nominal sampling position. The programs replace most of the manual data checking normally required.

THE PROGRAMS

STNDUP (Appendix 1)

This program checks up to 3000 sets of sample data for duplicated sample point numbers. If a survey number is also used, both the sample and survey numbers are tested for duplication. The data is read in 80 character records from logical unit 4 which must support both sequential and random access. The data for the duplicated numbers is output on logical unit 6 and spacing for better legibility is output on logical unit 3.

Data input from logical unit 4 is:

SURVNO, STATNO - format (I5, 1X, I4)
SURVNO (optional) - a four-figure survey number
STATNO - the data point number

CORDUP (Appendix 2)

This program checks the co-ordinates of up to 3000 sample points to locate all sample points less than a specified minimum distance from another sample point. The minimum acceptable distance is read from logical unit 5. The data is read in 80 character records from logical unit 4, which must support both sequential and random access. The data for the close sample points is output on logical unit 6, and spacing for better legibility is output on logical unit 3.

Control data read from logical unit 5 is:

DIST - the minimum acceptable separation of sample points in kilometres.

Data input from logical unit 4 is:

EAST, NORTH - format (10X, 2F9.1)
- the east and north co-ordinates of the sample point in metres. (EAST, NORTH in the range 0.0 to 700 000.0)

NOSTAT (Appendix 3)

This program reads up to 3000 sample point co-ordinates from logical unit 4 and then moves through a specified grid checking to see if there are any nominal sample points without an acceptably close actual sample point. The grid bounds, sample spacing, and maximum acceptable distance from the nominal grid points are read on logical unit 5. The co-ordinates of nominal sample points without a sample are output on logical unit 6.

Control data read in free format from logical unit 5 is:

XMIN, XMAX - the western and eastern boundaries of the grid (km)
YMIN, YMAX - the southern and northern boundaries of the grid (km)
XSP, YSP - the east separation and north separation of the nominal sample points (km)
DIST - the maximum acceptable distance of an actual sample point from a nominal sample point (km)

Data input from logical unit 4 is:

X, Y - format (10X, 2F9.1)
- the east and north co-ordinate of the sample point in metres. (X, Y in the range 0.0 to 700 000.0).

[11 July 1983]

APPENDIX 1

Program STNDUP

```
$TITL  STNDUP.FTN - DETECT DUPLICATED GRAVITY STATION NUMBERS
C DATA ON LU 4     N.B.  LU4 MUST BE DIRECT ACCESS
C OUTPUT OF DUPLICATES ON LU 6
C IF LU 6 IS PRINTER SET LU 3 TO THE PRINTER AS WELL
      INTEGER*2 ISURV(3000),ISTAT(3000)
      LOGICAL FOUND
      OPEN(UNIT=4,ACCESS='DIRECT')
C OPEN LU 4 FOR RANDOM ACCESS
      REWIND 4
      DO 10 I=1,3001
      READ(4,100,END=11) ISURV(I),ISTAT(I)
  100 FORMAT(I5,1X,I4)
   10 CONTINUE
      STOP 9998
C IF REACH HERE HAVE TOO MANY VALUES AND HAVE GONE OUTSIDE ARRAY BOUNDS
   11 NSTAT=I-1
C NO OF STATIONS
      MXSTAT=ISTAT(NSTAT)
      MNSTAT=MXSTAT
C WANT TO FIND THE RANGE OF STATION NUMBER SOMETIMES
      NL1=NSTAT-1
      DO 12 I=1,NL1
      IP1=I+1
      ISV=ISURV(I)
      IF (ISV .LT. 0) GOTO 12
C HAVE ALREADY FOUND THIS BATCH OF THE SAME NUMBERS
      IST=ISTAT(I)
      MXSTAT=MAX0(MXSTAT,IST)
      MNSTAT=MIN0(MNSTAT,IST)
      FOUND=.FALSE.
C ONLY WANT TO PRINT OUT ISTAT(I) ONCE
      DO 14 J=IP1,NSTAT
      IF (ISURV(J) .LT. 0) GOTO 14
C ALREADY FOUND THIS LOT OFDUPLICATES
      IF (IST .NE. ISTAT(J) .OR. ISV .NE. ISURV(J)) GOTO 14
C NOT SAME STATION NUMBER AND SURVEY NUMBER
      IF (FOUND) GOTO 15
C ALREADY FOUND ONE DUPLICATE
      WRITE(3,101)
  101 FORMAT(2X)
      CALL PRINT(I)
      FOUND=.TRUE.
   15 CALL PRINT(J)
```

```
C WRITE ALL DETAILS OF THE DUPLICATES
      ISURV(J)=-999
C NOTE THAT ALREADY DEALT WITH
   14 CONTINUE
   12 CONTINUE
      WRITE(3,102) MNSTAT,MXSTAT
  102 FORMAT(//' STATION NO.  RANGE ',2I5)
      ENDFILE 6
      STOP
      END
      SUBROUTINE PRINT(I)
      INTEGER*2 IA(40)
C RANDOM SEEK TO RECORD
      READ(4,100,REC=I) IA
  100 FORMAT(40A2)




      WRITE(6,100) IA
      RETURN
      END
```

APPENDIX 2

Program CORDUP

```
$TITL  CORDUP.FTN - LOCATE GRAVITY STATIONS LESS THAN A SET DISTANCE APART
C READ DISTANCE FORM LU 5
C INPUT DATA LU 4    N.B.   MUST BE RANDOM ACCESS
C OUTPUT DATA ON LU 6
C IF LU 6 IS THE PRINTER THEN ASSIGN LU 3 TO THE PRINTER
      LOGICAL FOUND
      REAL EAST(3000),ANORTH(3000)
      DOUBLE PRECISION DDIST,DX,DY,DSQ
      OPEN(UNIT=4,ACCESS='DIRECT')
C SET UP FOR RANDOM ACCESS
      OPEN(UNIT=5,FILE='CON:')
C OPEN THE CONSOLE FOR INPUT
      WRITE(5,100)
  100 FORMAT(' MIN.  DIST.  (KM)?')
      READ(5,*) DIST
      REWIND 4
      DO 10 I=1,3001
      READ(4,102,END=11) EAST(I),ANORTH(I)
  102 FORMAT(10X,2F9.1)
   10 CONTINUE
      STOP 9998
C IF GET TO HERE HAVE TOO MANY STATIONS FOR ARRAY BOUNDS
   11 NSTAT=I-1
C NUMBER OF READINGS
      DDIST=DIST*1000.
      DIST=DDIST
C CONVERT MIN.  DIST.  APART TO METRES
      DDIST=DDIST*DDIST
C SQUARE TO SAVE LATER DSQRT
      NL1=NSTAT-1
      DO 12 I=1,NL1
      IP1=I+1
      X=EAST(I)
      Y=ANORTH(I)
      XML=X-DIST
      XMP=X+DIST
      YML=Y-DIST
      YMP=Y+DIST
C ONLY WANT TO CHECK POINTS WITHIN A SQUARE OF SIDE 2*DIST
      FOUND=.FALSE.
      DO 14 J=IP1,NSTAT
      EJ=EAST(J)
      ANJ=ANORTH(J)
      IF (EJ .LE. XML .OR. EJ .GE. XMP .OR. ANJ .LE. YML
     . .OR. ANJ .GE. YMP) GOTO 14
```

```
C FAR ENOUGH AWAY
      DX=X-EJ
      DY=Y-ANJ
      DSQ=DX*DX+DY*DY
      IF (DSQ .GE.  DDIST) GOTO 14
C OUTSIDE THE MIN.  DIST.
      IF (FOUND) GOTO 15
C ALREADY ONE TOO CLOSE
      WRITE(3,200)
  200 FORMAT(2X)
      CALL PRINT(I,0.0D00)
      FOUND=.TRUE.
   15 CALL PRINT(J,DSQRT(DSQ)*0.001D00)




C PRINT OUT ALL THE DETAILS
   14 CONTINUE
   12 CONTINUE
      ENDFILE 6
      STOP
      END
      SUBROUTINE PRINT(I,DIST)
      INTEGER*2 IA(40)
      DOUBLE PRECISION DIST
      READ(4,100,REC=I) IA
  100 FORMAT(40A2)
      WRITE(6,101) IA,DIST
  101 FORMAT(40A2,F8.3)
      RETURN
      END
```

APPENDIX 3

Program NOSTAT

```
$TITL  NOSTAT.FTN - LOCATE NOMINAL POINTS WITHOUT A CLOSE STATION
C LU  4  INPUT
C LU  6  OUTPUT OF NOMINAL COORDS MISSING STATIONS
C READS (XMIN,YMIN),(XMAX,YMAX),SPACINGS AND ACCEPATBLE DIST FROM LU  5
      DIMENSION X(3000),Y(3000)
      DOUBLE PRECISION DXSP,DYSP,XX,YY,DDIST,DXK,DYK,XDIST
      OPEN(UNIT=5,FILE='CON:')
C SET THE CONSOLE AS LU 5 FOR INPUT
      REWIND 4
      WRITE(5,102)
  102 FORMAT(' XMINNNNNN XMAXXXXXX IN KM')
      READ(5,*) XMIN,XMAX
      WRITE(5,104)
  104 FORMAT(' YMINNNNNN YMAXXXXXX IN KM')
      READ(5,*) YMIN,YMAX
      WRITE(5,105)
  105 FORMAT(' XSPACEEEE YSPACEEEE IN KM')
      READ(5,*) XSP,YSP
      WRITE(5,106)
  106 FORMAT(' MAX.  DIST.  FROM POINT IN KM')
      READ(5,*) DIST
      DDIST=DIST
      DDIST=DDIST*1000.0D00
      DIST=DDIST
      XDIST=DDIST
C TO METRES AN STORE FOR LATER
      DDIST=DDIST*DDIST
C SQUARE IT
C NOW HAVE LIMITS AND SPACINGS IN KM
      XMIN=XMIN*1000.
      XMAX=XMAX*1000.
      YMIN=YMIN*1000.
      YMAX=YMAX*1000.
      XSP=XSP*1000.
      DXSP=XSP
      YSP=YSP*1000.
      DYSP=YSP
      XML=XMIN-DIST
      XMP=XMAX+DIST
      YML=YMIN-DIST
      YMP=YMAX+DIST
C GET DOUNDARY FOR POINTS TO FALL WITHIN IN METRES
      NX=((XMAX-XMIN)/XSP)+1.
      NY=((YMAX-YMIN)/YSP)+1.
C NO. OF GRID POINTS IN X AND Y DIRECTIONS
C
      N=0
    1 N=N+1
    2 READ(4,101,END=5) XN,YN
  101 FORMAT(10X,2F9.1)
      IF (XN .LT. XML .OR. XN .GT. XMP .OR. YN .LT. YML .OR.
     . YN .GT. YMP) GOTO 2
C PICK ONLY POINTS WITHIN THE AREA OF INTEREST
      X(N)=XN
      Y(N)=YN
      GOTO 1
    5 N=N-1
C N IS THE NUMBER OF DATA POINTS WITH COORDS X,Y
```

```
      YY=YMIN
      DO 10 I=1,NY
      XX=XMIN
      YM=YY-XDIST
      YP=YY+XDIST
C USE TO SEE IF IN SMALL RECTANGLE
      DO 20 J=1,NX
      XM=XX-XDIST
      XP=XX+XDIST
C USED TO SEE IF IN BASIC RECTANGLE
      DO 30 K=1,N
      XK=X(K)
      YK=Y(K)
      IF (XK .LT. XM .OR. XK .GT. XP .OR. YK .LT. YM .OR.
       YK .GT. YP) GOTO 30
C SEE IF POSSIBLY IN RANGE
      DXK=XK
      DYK=YK
C TO DOUBLE PRECISION
      DXK=DXK-XX
      DYK=DYK-YY
      IF (DXK*DXK+DYK*DYK .LE. DDIST) GOTO 21
C FOUND ONE WITHIN THE DESIRED DISTANCE
   30 CONTINUE
C IF FALL OUT DONT HAVE ANY IN DISTACE
      WRITE(6,200) XX,YY
  200 FORMAT(10X,2F9.0)
C WRITE THE COORDS NEEDED
   21 XX=XX+DXSP
   20 CONTINUE
      YY=YY+DYSP
   10 CONTINUE
      ENDFILE 6
      STOP
      END
```