



LEEDS
BECKETT
UNIVERSITY

Citation:

Wright, S and Webb, D (2004) Programmes used to run the time series analyses (APPENDIX B). Leeds Metropolitan University. (Unpublished)

Link to Leeds Beckett Repository record:

<https://eprints.leedsbeckett.ac.uk/id/eprint/2105/>

Document Version:

Other (Other)

The aim of the Leeds Beckett Repository is to provide open access to our research, as required by funder policies and permitted by publishers and copyright law.

The Leeds Beckett repository holds a wide range of publications, each of which has been checked for copyright and the relevant embargo period has been applied by the Research Services team.

We operate on a standard take-down policy. If you are the author or publisher of an output and you would like it removed from the repository, please [contact us](#) and we will investigate on a case-by-case basis.

Each thesis in the repository has been cleared where necessary by the author for third party copyright. If you would like a thesis to be removed from the repository or believe there is an issue with copyright, please contact us on openaccess@leedsbeckett.ac.uk and we will investigate on a case-by-case basis.

Appendix B:

PROGRAMMES USED TO RUN THE TIME SERIES ANALYSES

\\LISTING OF :RLXXX.TAD1UNIV(8/) PRODUCED ON 29OCT80 AT 12.41.30

\\OUTPUT ON UMRCC 6A BY ':RLXXX.TAD1MOP' ON 9MAR83 AT 17.34.45 USING I186

DOCUMENT TAD1UNIV

```
PROGRAM PS(INPUT,OUTPUT,IRD,TAPE5=IRD,TAPE2=INPUT,TAPE6=OUTPUT)
DIMENSION LPHA(20),S(132),S1(132),S2(132)
DIMENSION PLOT(132,4),A1(132),A2(132)
DIMENSION IJK(132)
DO690 J=1,132
S1(J)=0.
S2(J)=0.
S(J)=0.
A1(J)=0.
A2(J)=0.
690 CONTINUE
READ(5,10)JCOUNT,LS,LSD
98 CONTINUE
JC=1
99 READ(5,9)(LPHA(J),J=1,20)
LS=3
READ(5,10)N,NT,NVA
READ(5,10)NV,NE,NSD,NED,NTI,NC,(IJK(J),J=1,N)
IF(NV .GT. 500)GOTO101
L=0
LZ=0
535 L=L+1
IF(IJK(L) .GT. 0)LZ=1
IF(LZ .EQ. 1)GOTO536
IF(L .LT. N)GOTO535
536 CONTINUE
WRITE(6,7)NVA
WRITE(6,10)NV,NE,NSD,NED,NTI,NC,(IJK(J),J=1,N)
IF(NT .LT. 0)NT= -NT
N=N+1
12 MM=N+11
READ(5,13)(IJK(J),J=N,MM)
IF(LZ .EQ. 1)GOTO537
538 L=L+1
IF(IJK(L) .GT. 0)LZ=1
IF(LZ .EQ. 1)GOTO537
IF(L .LT. MM)GOTO538
537 CONTINUE
WRITE(6,13)(IJK(J),J=N,MM)
N=N+12
IF(NT .GT. N)GOTO12
NK=0
NMISS=L
DO823 J=NMISS,MM
NK=NK+1
823 S(NK)=FLOAT(IJK(J))
NT=NT-NMISS+1
11 WRITE(6,6)
WRITE(6,9)(LPHA(J),J=1,20)
101 N1=1
WRITE(6,1)LS
NS=1
103 IF((LS+NS) .LE. NT)GOTO100
IF(NUM .EQ. 1)GOTO98
WRITE(6,14)
```

```

CALL MINMX(A1,NUM)
WRITE(6,15)
CALL MINMX(A2,NUM)
DO750 J=1,NUM
PLOT(J,3)=A1(J)
PLOT(J,4)=A2(J)
750 CONTINUE
WRITE(6,9)(LPHA(J),J=1,20)
CALL PLOT(PLOT,NUM,4,U.,100.)
WRITE(6,6)
LS=LS+LSD
IF(LS .LT. NT)GOTO101
GOTO99
100 N3=N1+1
K=N1
DO200 J=1,LS
S1(J)=S(K)
200 K=K+1
190 K=N3
DO201 J=1,LS
S2(J)=S(K)
201 K=K+1
SD1=0.
SD2=0.
SM1=0.
SM2=0.
COR=0.
DO 203 J=1,LS
SM1=SM1+S1(J)
SM2=SM2+S2(J)
SD1=SD1+S1(J)**2.
SD2=SD2+S2(J)**2.
203 COR=COR+S1(J)*S2(J)
SM1=SM1/LS
SM2=SM2/LS
IF(((SD1/LS)-SM1**2.) .LE. 0.)GOTO30
IF(((SD2/LS)-SM2**2.) .LE. 0.)GOTO30
SD1=SQRT((SD1/LS)-SM1**2.)
SD2=SQRT((SD2/LS)-SM2**2.)
COR=((COR/LS)-SM1*SM2)/(SD1*SD2)
GOTO31
30 COR=0.
31 CONTINUE
NUM=N1
PLOT(N1,JC)=(COR*50)+50
A1(N1)=SM1
IF(JC .EQ. 1)A2(N1)=SM2
JC=JC+1
IF(JC .GT. JCOUNT)GOTO110
N3=N3+1
GOTO190
110 NS=NS+1
N1=N1+1
JC=1
GOTO103
999 CONTINUE
STOP
1 FORMAT(15H SERIES LENGTH ,I3)
6 FORMAT(///)
7 FORMAT(25H DATA CARDS FOR VARIABLE ,I3,16H LOOK LIKE THIS //)
9 FORMAT(20A4)
10 FORMAT(2I3,2I4,I3,I2,12I5)
13 FORMAT(19X,12I5)
14 FORMAT(11H FOR MEAN1 /19H ACTUAL VALUES ARE )
15 FORMAT(11H FOR MEAN2 /19H ACTUAL VALUES ARE )
END

```



```

SUBROUTINE PLOT(Y,NN,II,BB,AA)
DIMENSION Y(132,4),A(101),P(6)
DATA P/1H1,1H2,1HX,1H+,1H-,1HI/
YI=BB
YA=AA
IF(YA-YI)1,1,4
1 YI=Y(1,1)
YA=YI
DO 3 I=1,II
DO2 N=1,NN
IF(Y(N,I) .GT. YA) YA=Y(N,I)
IF(Y(N,I) .LT. YI) YI=Y(N,I)
2 CONTINUE
3 CONTINUE
4 SCL=100./(YA-YI)
WRITE(6,100)YI,YA
100 FORMAT(21H0 EXTREMES OF SCALE ,E12.4,3H TO,E12.4//)
M=0
GOTO50
5 M=1
DO20 N=1,NN
DO 10 I=2,100
10 A(I)=P(5)
A(1)=P(4)
A(51)=A(1)
A(101)=A(1)
DO15 I=1,II
IY=IFIX(SCL*(Y(N,I)-YI))
IF(IY*(101-IY)) 15,14,14
14 A(IY+1)=P(I)
15 CONTINUE
I HAVE CHANGED PLOT HERE TO PRINT OUT TWO CORRELATIONS RATHER THAN Y(N,1)
IN STATEMENT 20. TAKE OUT ALL CARDS FROM HERE TO FORMAT101 AND REPLACE
20 BY 20 WRITE(6,101)N,Y(N,1),A AND STATEMENT 101 FORMAT(I4,2X,F12.4,
1X,101A1
ICOR1=(Y(N,1)-50)*2
ICOR2=(Y(N,2)-50)*2
20 WRITE(6,101)N,ICOR1,ICOR2,A
101 FORMAT(I4,2X,2I6,1X,101A1)
50 DO51 I=2,100
51 A(I)=P(4)
DO52 I=1,101,5
52 A(I)=P(6)
WRITE(6,102)A
ALSO REPLACE 102 BY 102 FORMAT(19X,101A1)
102 FORMAT(19H TIME CORT1 CORT2 ,101A1)
IF(M) 55,5,55
55 RETURN
END
SUBROUTINE MINMX(A,N)
DIMENSION A(132)
WRITE(6,202)(A(J),J=1,N)
AMX = A(1)
AMN = A(1)
DO 100 J=1,N
IF(A(J) .GT. AMX)AMX=A(J)
IF(A(J) .LT. AMN)AMN= A(J)
100 CONTINUE
200 FORMAT(18H MAXIMUM VALUE OF ,F12.3,16H AND MINIMUM OF ,F12.3)
201 FORMAT(19H SCALED VALUES ARE )
202 FORMAT(12F6.0)
DO101 J=1,N
101 A(J)=(A(J)-AMN)*100/(AMX-AMN)
WRITE(6,200)AMX,AMN
WRITE(6,201)
WRITE(6,202)(A(J),J=1,N)
RETURN
END

```

 PRINTING OF :RLXXX,TAD1IRISHP(367) PRODUCED ON 20MAR80 AT 17.56.34
 PRINT ON BURCC 6A BY :RLXXX,TAD1IRISHP ON 20MAR80 AT 17.32.19 USING 1186
 PRINT - TAD1IRISHP

```

PROGRAM IRICINPUT,OUTPUT,IRISHP,TAPE2=IRISHP,TAPE5=INPUT
  >TAPE6=OUTPUT)
  DIMENSION LPHA(20),IJK(100),S(100),S1(100),S2(100),TCOR(100),
  >R(100),R1(100),M2(100),COR(100),COR1(100),COR2(100),MCCR(100),
  >S0(100),S0(100),PLOTX(100,5)
  CONTINUE
  NV1=0
  NV2=0
  NN=1
  READ(2,1)NV1,NV2,LAG,LS,LS0,LLS
  CONTINUE
10 READ(2,2)(LPHA(J),J=1,20)
  READ(2,3)(NV,(IJK(J),J=1,12))
  IF(NV .EQ. NV1)GOTO101
  IF(NV .EQ. NV2)GOTO102
  GO103 J=1,7
11 READ(2,2)(LPHA(K),K=1,20)
  GO104
12 NV1=1
  N=13
  WRITE(6,2)(LPHA(J),J=1,20)
  WRITE(6,3)(NV,(IJK(J),J=1,12))
13 N=N+1
  READ(2,4)(IJK(J),J=N,N+1)
  WRITE(6,4)(IJK(J),J=N,N+1)
  N=N+1
  IF(N .LT. 94)GOTO105
  GO106 J=1,96
14 R(J)=FLOAT(IJK(J))
  WRITE(6,910)(R(J),J=1,96)
15 FORMAT(12F9,2)
  GO104
16 NV2=1
  N=13
  WRITE(6,2)(LPHA(J),J=1,20)
  WRITE(6,3)(NV,(IJK(J),J=1,12))
17 N=N+1
  READ(2,4)(IJK(J),J=N,N+1)
  WRITE(6,4)(IJK(J),J=N,N+1)
  N=N+1
  IF(N .LT. 94)GOTO106
  GO109 J=1,96
18 S(J)=FLOAT(IJK(J))
  WRITE(6,910)(S(J),J=1,96)
19 CONTINUE
  IF(NV1=NV2) .LT. 13)GOTO110
20 I=3
21 I=I+LS-1
  I=I-1
  I=I-1
  I=I-1
  I=I-1
  I=I-1
  IF(I=96)GOTO112
  I=0
  GO112 J=I+I
  
```

^*****
 \LISTING OF :RLXXX.TAD1IRISHP(36/) PRODUCED ON 20MAR80 AT 17.36.34
 \OUTPUT ON UMRCC 6A BY ':RLXXX.TAD1HOP' ON 9MAR83 AT 17.32.19 USING I186
 DOCUMENT TAD1IRISHP

```

PROGRAM IRI(INPUT,OUTPUT,IRISHD,TAPE2=IRISHD,TAPE5=INPUT
+,TAPE6=OUTPUT)
  DIMENSION LPHA(20),IJK(100),S(100),S1(100),S2(100),TCOR(100),
+,R(100),R1(100),R2(100),COR(100),COR1(100),COR2(100),WCOR(100),
+,SO(100),PO(100),PLOT(100,5)
887 CONTINUE
  MV1=0
  MV2=0
  NN=1
  READ(2,1)NV1,NV2,LAG,LS,LSD,LLS
  CONTINUE
110 READ(2,2)(LPHA(J),J=1,20)
  READ(2,3)(NV,(IJK(J),J=1,12))
  IF(NV .EQ. NV1)GOTO101
  IF(NV .EQ. NV2)GOTO102
  DO103 J=1,7
103 READ(2,2)(LPHA(K),K=1,20)
  GOTO104
101 MV1=1
  N=13
  WRITE(6,2)(LPHA(J),J=1,20)
  WRITE(6,3)(NV,(IJK(J),J=1,12))
105 MM=N+11
  READ(2,4)(IJK(J),J=N,MM)
  WRITE(6,4)(IJK(J),J=N,MM)
  N=N+12
  IF(N .LT. 94)GOTO105
  DO106 J=1,96
106 R(J)=FLOAT(IJK(J))
  WRITE(6,910)(R(J),J=1,96)
910 FORMAT(12F9.2)
  GOTO104
102 MV2=1
  N=13
  WRITE(6,2)(LPHA(J),J=1,20)
  WRITE(6,3)(NV,(IJK(J),J=1,12))
108 MM=N+11
  READ(2,4)(IJK(J),J=N,MM)
  WRITE(6,4)(IJK(J),J=N,MM)
  N=N+12
  IF(N .LT. 94)GOTO108
  DO109 J=1,96
109 S(J)=FLOAT(IJK(J))
  WRITE(6,910)(S(J),J=1,96)
104 CONTINUE
  IF((MV1*MV2) .LT. 1)GOTO110
151 ISP=3
111 IEP=ISP+LS-1
  ISP1=ISP-1
  ISP2=ISP-2
  IEP1=IEP-1
  IEP2=IEP-2
  IF(IEP .GT. 96)GOTO999
  K=0
  DO112 J=ISP,IEP

```



```

      K=K+1
      S0(K)=S(J)
112  R0(K)=R(J)
      K=0
      D0113 J=ISP1,IEP1
      K=K+1
      S1(K)=S(J)
113  R1(K)=R(J)
      K=0
      D0114 J=ISP2,IEP2
      K=K+1
      S2(K)=S(J)
114  R2(K)=R(J)
      CALL KOR(LS,R2,S0,C)
      TCOR(NN)=C
      CALL KOR(LS,R1,S0,C)
      WCOR(NN)=C
      CALL KOR(LS,R0,S0,C)
      COR(NN)=C
      CALL KOR(LS,R0,S1,C)
      COR1(NN)=C
      CALL KOR(LS,R0,S2,C)
      COR2(NN)=C
      ISP=ISP+1
      NN=NN+1
      GOT0111
999  CONTINUE
      NN=NN-1
      WRITE(6,957)LS
      WRITE(6,958)
958  FORMAT(62H VAR A LAG 2 VAR A LAG 1 INTERACTION VAR B LAG 1 VAR B
+ LAG 2 )
957  FORMAT(18H SERIES LENGTH ,I4)
      D0150 J=1,NN
      WRITE(6,911)NN,TCOR(J),WCOR(J),COR(J),COR1(J),COR2(J)
      PLOT(J,3)=COR(J)
      PLOT(J,2)=WCOR(J)
      PLOT(J,4)=COR1(J)
      PLOT(J,1)=TCOR(J)
      PLOT(J,5)=COR2(J)
150  CONTINUE
      D0161 J=1,NN
      D0152 K=1,5
      IF(PLOT(J,K) .LT. 0.)GOT0155
      IF(PLOT(J,K) .LT. .5)PLOT(J,K)=0.
      IF(PLOT(J,K) .EQ. 0.)GOT0152
      PLOT(J,K)=PLOT(J,K)-.5
      GOT0152
155  IF(PLOT(J,K) .GT. -.5)PLOT(J,K)=0.
      IF(PLOT(J,K) .EQ. 0.)GOT0152
      PLOT(J,K)=PLOT(J,K)+.5
152  CONTINUE
161  CONTINUE
      WRITE(6,957)LS
      CALL PLOT(PLOT,NN,5,-.5,.5)
      NN=1
      LS=LS+LSD
      IF(LS .LE. LLS)GOT0151
      CONTINUE
      STOP
1  FORMAT(6I2)
2  FORMAT(20A4)
3  FORMAT(15X,12,2X,12I5)
4  FORMAT(19X,12I5)
911  FORMAT(14,5F12.4)
      END

```



```

SUBROUTINE KOR(NK,S1,S2,COR)
DIMENSION S1(100),S2(100)
SD1=0.
SD2=0.
SM1=0.
SM2=0
COR=0.
DO100 J=1,NK
SM1=SM1+S1(J)
SM2=SM2+S2(J)
SD1=SD1+S1(J)**2.
SD2=SD2+S2(J)**2.
100 COR=COR +S1(J)*S2(J)
SM1=SM1/NK
SM2=SM2/NK
X=(SD1/NK) -SM1**2.
Y=(SD2/NK) -SM2**2.
IF(X .LE. 0.)GOTO101
IF(Y .LE. 0.)GOTO101
SD1=SQRT(X)
SD2=SQRT(Y)
COR=((COR/NK) - SM1*SM2)/(SD1*SD2)
GOTO102
101 COR=0.
102 CONTINUE
RETURN
END
SUBROUTINE PLOT(Y,NN,II,BB,AA)
DIMENSION Y(100,5),A(101),P(6)
DATA P/1H1,1H2,1H3,1H4,1H5,1H /
YI=BB
YA=AA
IF(YA-YI)1,1,4
1 YI=Y(1,1)
YA=YI
DO 3 I=1,II
DO 2 N=1,NN
IF(Y(N,I) .GT. YA)YA=Y(N,I)
IF(Y(N,I) .LT. YI)YI=Y(N,I)
2 CONTINUE
3 CONTINUE
4 SCL=100./(YA-YI)
WRITE(6,100)YI,YA
100 FORMAT(21H0 EXTREMES OF SCALE ,E12.4,3H TO,E12.4//)
M=0
GOTO50
5 M=1
DO20 N=1,NN
DO10 I=2,100
10 A(I)=P(6)
A(1)=P(4)
A(51)=A(1)
A(101)=A(1)
DO 15 I=1,II
IY=IFIX(SCL*(Y(N,I)-YI))
IF(IY*(101-IY))15,14,14
14 A(IY+1)=P(I)
15 CONTINUE
ICOR1= IFIX(100*Y(N,2)) +50
ICOR2= IFIX(100*Y(N,3)) +50
ICOR3= IFIX(100*Y(N,4)) +50
20 WRITE(6,101)N,ICOR1,ICOR2,ICOR3,A
101 FORMAT(I3,3I5,1X,101A1)
50 DO51 I=2,100
51 A(I)=P(4)
DO52 I=1,101,5

```

```
52 A(I)=P(1)
   WRITE(6,102)A
102 FORMAT(19H TP AL1 INT BL1 ,101A1)
   IF(M)55,5,55
55 RETURN
   END
```

^*****

LISTING OF IRLXXX.TAOTIRISHMP5(487) PRODUCED ON TOJUN82 AT 14.20.17
 INPUT ON UNBCC 64 BY IRLXXX.TAOTIRISHMP5 ON 09/08/82 AT 17.48.30 USING 1186
 DOCUMENT TABIRISHMP5

```

PROGRAM MULT(INPUT,OUTPUT,IP339A,TAPES=IRI5DA,TAPE2=INPUT,
+TAPES=OUTPUT)
  COMMON LPHA(20)
  COMMON/ABC/ S(150),ST(150),S2(150),IJK(150),R(150)
+DUM(150)
  COMMON/PR1/DATA(16,150),COR(316,114),STORE(114,114)
  COMMON/XXX/AT(114),BT(114),NVV(114),IKK(114)
+KTYP(50),MPP(114)
  COMMON/ZZZ/ AA(114),MP(114)
  DIMENSION V(150),Z(150)
  LEVEL 2,DATA,COR,STORE,S,ST,S2,IJK,R,DUM,LPHA
+AT,BT,NVV,KTYP,NRP,AA,MP,IKK
  DO99J=1,114
  DO98K=1,114
  COR(J,K)=0.
  STORE(J,K)=0.
98 CONTINUE
99 CONTINUE
  DO100 J=1
  ST(J)=0.
  S2(J)=0.
  S(J)=0.
  IJK(J)=0.
  R(J)=0.
  DUM(J)=0.
100 CONTINUE
COUNT IS LAA*LB IS SERIES LENGTH*LSO IS INCREMENT FOR SUBSEQUENT SERIES
NO NVS IS NUMBER OF VARIABLES, WHICH IS 20 MAXIMUM UNLESS YOU ALTER DATA
COR TO ALLOW FOR TIME LAB OF UP TO TWO AS AT PRESENT.
NOTE ALL VARIABLES MUST BE SAME NUMBER OF TIME POINTS
N1 IS NUMBER OF TIME POINTS ON FIRST CARD, N2 ON SUBSEQUENT CARDS
NT IS TOTAL NUMBER OF TIME POINTS, NVA IS VARIABLE IDENTIFICATION NUMBER
  READ(5,1)LS,LSO,JSP,JEP
  READ(5,1)NVS,NT,N,N2,JCOUNT
  NVAL=JSP-JCOUNT
  DO101 IJ=1,NVS
  JNN=N
  READ(5,2)(LPHA(I),I=1,20)
  WRITE(6,2)(LPHA(I),I=1,20)
  READ(5,3)MPP,NE,NNU,NEO,NTI,6C,(IJK(I),I=1,N)
  N=N+1
102 N=N+N2-1
  READ(5,4)(IJK(I),I=N,N2)
  N=N+N2
  IF(NT .GT. 1)GO TO102
  DO105 K=1,N
  S(K)=FLOAT(IJK(K))
  THE DATA IS READ AND REPRODUCED FOR EACH VARIABLE BY ABOVE SECTION
  IT IS STORED IN THE ARRAY CALLED DATA
  DATA(IJ,K)=S(K)
103 CONTINUE
  N=N+N
104 CONTINUE
  DO820 I=1,150
  IJK(I)=J
  
```

^*****

\LISTING OF :RLXXX.TAO1IRISHP5(48/) PRODUCED ON 10JUN82 AT 14.26.17
\OUTPUT ON UMRCC 6A BY ':RLXXX.TAO1MOP' ON 9MAR83 AT 17.48.30 USING I186
DOCUMENT TAO1IRISHP5

```
PROGRAM MULT(INPUT,OUTPUT,IRISDA,TAPE5=IRISDA,TAPE2=INPUT,  
+TAPE6=OUTPUT)  
COMMON LPHA(20)  
COMMON/ABC/ S(150),S1(150),S2(150),IJK(150),P(150)  
+, DUM(150)  
COMMON/PR1/DATA(38,150),COR(114,114),STORE(114,114)  
COMMON/XXX/A1(114),B1(114),NVV(114),IKK(114)  
+,KTYP(50),MPP(114)  
COMMON/ZZZ/ AA(114),MP(114)  
DIMENSION Y(150),Z(150)  
LEVEL 2,DATA,COR,STORE,S,S1,S2,IJK,R,DUM,LPHA  
+,A1,B1,NVV,KTYP,MPP,AA,MP,IKK  
D099J=1,114  
D098K=1,114  
COR(J,K)=0.  
STORE(J,K)=0.  
98 CONTINUE  
99 CONTINUE  
D0100 J=1,150  
S1(J)=0.  
S2(J)=0.  
S(J)=0.  
IJK(J)=0.  
R(J)=0.  
DUM(J)=0.  
100 CONTINUE  
CJCOUNT IS LAG,LS IS SERIES LENGTH, LSD IS INCREMENT FOR SUBSEQUENT SERIES  
CAND NVS IS NUMBER OF VARIABLES, WHICH IS 20 MAXIMUM UNLESS YOU ALTER DATA  
C COR TO ALLOW FOR TIME LAG OF UP TO TWO AS AT PRESENT.  
C NOTE ALL VARIABLES MUST BE SAME NUMBER OF TIME POINTS  
C N1 IS NUMBER OF TIME POINTS ON FIRST CARD, N2 ON SUBSEQUENT CARDS,  
C NT IS TOTAL NUMBER OF TIME POINTS, NVA IS VARIABLE IDENTIFICATION NUMBER  
READ(5,1)LS,LSD,JSP,JEP  
READ(5,1)NVS,NT,N,N2,JCOUNT  
NTSL=JSP-JCOUNT  
D0101 IJ=1,NVS  
NNN=N  
READ(5,2)(LPHA(J),J=1,20)  
WRITE(6,2)(LPHA(J),J=1,20)  
READ(5,4)NV,NE,NSD,NED,NTI,NC,(IJK(J),J=1,N)  
N=N+1  
102 MM=N+N2-1  
READ(5,6)(IJK(J),J=N,MM)  
N=N+N2  
IF(NT .GT. N)GOTO102  
D0103 K=1,MM  
S(K)=FLOAT(IJK(K))  
C THE DATA IS READ AND REPRODUCED FOR EACH VARIABLE BY ABOVE SECTION  
C IT IS STORED IN THE ARRAY CALLED DATA  
DATA(IJ,K)=S(K)  
103 CONTINUE  
N=NNN  
101 CONTINUE  
D0820J=1,150  
820 IJK(J)=J
```



```

C THE NEXT BIT BUILDS UP CORRELATION MATRIX FOR EACH CUT OF 3 TIME POINTS
C IS LAGS GREATER THAN TWO ARE USED THEN COR NEEDS TO BE EXPANDED FROM 3*
C ISP IS STARTING TIME POINT AND IEP IS ENDING POINT FOR EACH COR
  NVSM=NVS-1
  ISP=JSP
998 IEP=ISP+LS-1
  D0700 J=1,NVS
  NNN=(J*3)-2
  MM=0
  D0701 K=ISP,IEP
  MM=MM+1
  R(MM)=DATA(J,K)
701 CONTINUE
C C THIS SETS FIRST VARIABLE IN R. BELOW SETS OTHERS
  ISP1=ISP-1
  ISP2=ISP-2
  IEP1=IEP-1
  IEP2=IEP-2
C ISP1 IS START OF LAG ONE SERIES, 2 OF LAG TWO
C SIMILARLY FOR END OF SERIES IEP
C IF THE LAG TWO END IS .GT. NUMBER OF TIME POINTS NT THEN END
  IF(IEP .GT. NT)GOTO999
  IF(IEP .GT. JEP)GOTO999
  N=0
  D0704 J2=1,NVS
  L=0
  D0705 K=ISP,IEP
  L=L+1
705 S(L)=DATA(J2,K)
  L=0
  D0706 K1=ISP1,IEP1
  L=L+1
706 S1(L)=DATA(J2,K1)
  L=0
  D0707 K2=ISP2,IEP2
  L=L+1
707 S2(L)=DATA(J2,K2)
C HAVING SET UP S,S1,S2 FOR EACH OF NEXT VARIABLES COORELATE
  D060 L=1,LS
  Y(L)=R(L)
60 Z(L)=S(L)
  CALL KOR(LS,Y,Z,CORR)
  N=N+1
  IF(J2 .EQ. J)CORP=0.
  DUM(N)=CORR
C STACK AWAY CORR IN DUM
  D061 L=1,LS
61 Z(L)=S1(L)
  CALL KOR(LS,Y,Z,CORR)
  N=N+1
  DUM(N)=CORR
  D062L=1,LS
62 Z(L)=S2(L)
  CALL KOR(LS,Y,Z,CORR)
  N=N+1
  DUM(N)=CORR
704 CONTINUE
  NNP=NNN+1
C NOW STACK DUM IN COR BEFORE GOING BACK TO UP J BY 1
  D0708 M=1,N
  COR(NNN,M)=DUM(M)
  COR(M,NNN)=DUM(M)
708 CONTINUE
700 CONTINUE
  WRITE(6,12)LS
  WRITE(6,7)NTSL

```

```

CALL TY(NTSL,N)
NTSL=NTSL+LSD
C THE NEXT BIT OUTPUTS THIS TIME SLICE
ISP=ISP+LSD
GOTO998
999 CONTINUE
C THE NEXT5 LINES SHOULD HAVE COLUMN 1 C REMOVED FOR
C RUNNING ON A LINE PRINTER THAT WILL ALLOW SEVERAL SERIES
C OUTPUT TO BE PRINTED ALL AT ONCE. AT PRESENT ONE SERIES
C USES ABOUT 7000 TO 8000 LINES OF OUTPUT.
C IF((LS+LSD) .GT. NT)GOTO997
C NTSL=1
C LS=LS+LSD
C ISP=JSP
C GOTO998
997 CONTINUE
STOP
1 FORMAT(7I3)
2 FORMAT(20A4)
3 FORMAT(16H DATA CARDS FOR ,20A4 )
4 FORMAT(2I3,2I4,I3,I2,12I5)
5 FORMAT(10H VARIABLE INT ,I3)
6 FORMAT(19X,12I5)
7 FORMAT(17H FOR TIME PERIOD ,I6)
8 FORMAT(I4,15F7.2)
9 FORMAT(///)
10 FORMAT(I4,3F7.2)
11 FORMAT(8H VARTINE ,I3,14I7)
12 FORMAT(50H OUTPUT FOR TIME SERIES CLUSTER FOR SERIES LENGTH ,I5/)
END
SURROUTINE KOR(NK,PS1,PS2,COR)
DIMENSION PS1(150),PS2(150)
SD1=0.
SD2=0.
SM1=0.
SM2=0.
COR=0.
DO100 J=1,NK
SM1=SM1+PS1(J)
SM2=SM2+PS2(J)
SD1=SD1+PS1(J)**2.
SD2=SD2+PS2(J)**2.
100 COR=COR+PS1(J)*PS2(J)
SM1=SM1/NK
SM2=SM2/NK
X=(SD1/NK) - SM1**2.
Y=(SD2/NK) - SM2**2.
IF(X .LE. 0.)GOTO101
IF(Y .LE. 0.)GOTO101
SD1=SQRT(X)
SD2=SQRT(Y)
COR=((COR/NK) - SM1*SM2)/(SD1*SD2)
IF(COR .GT. .999)COR=.999
IF(COR .LT. -.999)COR=-.999
GOTO102
101 COR=0.
102 CONTINUE
RETURN
END
SURROUTINE TY(NTSL,N)
COMMON LPHA(20)
COMMON/XXX/ A(114),B(114),NV(114),IK(114)
+KTP(50),MP(114)
COMMON/PR1/DATA(38,150),COR(114,114),STORE(114,114)
DIMENSION X(114)
DIMENSION LV(150)

```

```

LEVEL 2,A,B,NV,IK,LPHA,KTYP,MP,DATA,COR,STORE
AMX=0.
INT=0
DO100J=1,N
DO101K=1,N
IK(K)=0
LV(K)=0
101 X(K)=COR(J,K)
CALL COLMX(AMX,X,N,INT,MP)
MP(J)=MP
B(J)=AMX
A(J)=0.
NV(J)=INT
DO 102 K=1,N
102 COR(J,K)=0.0
100 CONTINUE
NTYPE=1
LEV=1
103 CONTINUE
DO104 K=1,N
60 X(K)=B(K)
CALL COLMX(AMX,X,N,INT,MP)
COMMENT THIS SETS CUT OF POINT AT .5
IF(AMX .LT. .5)GOTO998
COR(LEV,INT)=AMX*MP(INT)
IMT=NV(INT)
COR(LEV,IMT)=AMX*MP(IMT)
B(INT)=0.
B(IMT)=0.
NV(INT)=0.
NV(IMT)=0.
LEV=LEV+1
105 ICOUNT=0
DO110 J=1,N
IF(NV(J) .EQ. INT)GOTO650
IF(NV(J) .EQ. IMT)GOTO650
GOTO110
COMMENT THIS SETS CUT OFF AT .5
650 IF(B(J) .LT. .5)GOTO651
COR(LEV,J)=(NV(J)+B(J))*MP(J)
ICOUNT=ICOUNT+1
651 B(J)=0.
NV(J)=0
110 CONTINUE
IF(ICOUNT .EQ. 0)GOTO999
115 ICOUNT=0
DO111 J=1,N
IF(COR(LEV,J) .NE. 0)IK(J)=J
111 CONTINUE
LEV=LEV+1
DO112 J=1,N
IF(IK(J) .EQ. 0)GOTO112
DO113 K=1,N
IF(NV(K) .NE. IK(J))GOTO113
IF(B(K) .LT. .5)GOTO656
COR(LEV,K)=(NV(K)+B(K))*MP(K)
ICOUNT=ICOUNT+1
656 B(K)=0.
NV(K)=0
113 CONTINUE
112 CONTINUE
IF(ICOUNT .EQ. 0)GOTO999
GOTO115
999 NTYPE=NTYPE+1
KTYP(NTYPE)=LEV
LEV=LEV+1

```



```

GOTO103
998 CONTINUE
  J=1
345 CONTINUE
  WRITE(6,13)
  L=0
  DO810 J=1,LEV
  DO801 K=1,N
  IF(COR(J,K) .EQ. 0.)GOTO801
  L=L+1
  IF(COR(J,K) .GT. 1.)GOTO802
  IF(COR(J,K) .LT. -1.)GOTO802
  IF(COR(J,K) .GT. 0.)A(L)=COR(J,K) +K
  IF(COR(J,K) .LT. 0.)A(L)=COR(J,K)-K
  GOTO801
802 A(L)=COR(J,K)
  LV(L)=K
801 CONTINUE
  IF(L .EQ. 0)WRITE(6,15)
  IF(L .EQ. 0)GOTO800
  IF(L .LE. 10)WRITE(6,14)(J,L,(A(I),I=1,L))
  IF(L .LE. 10)WRITE(6,16)(LV(I),I=1,L)
  IF(L .LE. 10)GOTO800
  N1=1
  N2=10
803 WRITE(6,14)(J,L,(A(I),I=N1,N2))
  WRITE(6,16)(LV(I),I=N1,N2)
  N1=N1+10
  N2=N2+10
  IF(N1 .GT. L)GOTO800
  IF(N2 .GE. L)N2=L
  GOTO803
800 L=0
  DO815 IN=1,N
815 LV(IN)=0
810 CONTINUE
  DO320 J=1,N
  DO320K=1,N
320 COR(J,K)=0.
  1 FORMAT(I3)
  2 FORMAT(10F8.4)
  3 FORMAT(20F6.3)
  4 FORMAT(10F6.3)
  13 FORMAT(20H MULT TYPAL START )
  14 FORMAT(I4,I4,2X,10F9.3)
  15 FORMAT(10H )
  16 FORMAT(10X,10(5X,I4))
  RETURN
  END
  SUBROUTINE COLMX(AMX,A,N,K,NP)
  DIMENSION A(114)
  COMMON/ZZZ/ AAA(114),MP(114)
  LEVEL 2,AAA,MP
  AMAX=0.
  DO100J=1,N
  MP(J)=1
  IF(A(J) .LT. 0)MP(J)=-1
  A(J)=A(J)*MP(J)
  IF(A(J) .GT. AMAX)K=J
  CONTINUE
100 IF(A(J) .GT. AMAX)AMAX=A(J)
  AMX=AMAX
  NP=MP(K)
  IF(AMX .EQ. 0.)K=0
  IF(K .EQ. 0)NP=0
  RETURN
  END

```