Journal of Modern Applied Statistical Methods

Volume 8 | Issue 2 Article 32

11-1-2009

JMASM29: Dominance Analysis of Independent Data (Fortran)

Du Feng
Texas Tech University, du.feng@ttu.edu

Normal Cliff
University of Southern California, nrcliff5@q.com

Journal of Modern Applied Statistical Methods
November 2009, Vol. 8, No. 2, 659-669

1538 – 9472/09/$95.00

JMASM29: Dominance Analysis of Independent Data (Fortran)

Du Feng
Texas Tech University

Norman Cliff
University of Southern California

A Fortran 77 program is provided for an ordinal dominance analysis of independent two-group comparisons. The program calculates the ordinal statistic, *d*, and statistical inferences about δ. The source codes and an executable file are available at http://www.depts.ttu.edu/hdfs/feng.php.

Key words: ordinal statistic, dominance analyses, independent *d*, Fortran.

Introduction

The frequently encountered location comparison problem in behavioral and psychological research is usually answered by the two-sample *t*-test, comparing means of the two groups, or the parallel one-way ANOVA. However, it has been argued that ordinal alternatives to mean comparisons, such as the dominance analysis δ (Agresti, 1984; Cliff, 1991, 1993; Hettmansperger, 1984; Randles & Wolfe, 1979), have advantages over the classical ones, because data in the social sciences are often ordinal in nature. In addition, ordinal methods are invariant under monotonic transformation, and can be more robust than the traditional normal-based statistics methods when the parametric assumptions are violated (Caruso & Cliff, 1997; Cliff, 1993; Long, Feng, & Cliff, 2003). This dominance analysis, δ, is summarized by the ordinal statistic *d* which compares the proportion of times a score from one group or under one condition is higher than a score from the other, to the proportion of times when the reverse is true. The *d* method not only tests the H_0: δ = 0, but also allows for determination of confidence interval (CI) bounds.

Fligner and Policello (1981) introduced a robust version of the Wilcoxon-Mann-Whitney test (Mann & Whitney, 1947) for comparing the medians of two independent continuous distributions, and tested behavior of *d*, using the sample estimate of its variance. Cliff (1993) suggested a modification of Fligner and Policello's (1981) procedure by deriving an unbiased sample estimate of the variance of *d* and setting a minimum allowable value for it in order to increase the efficiency of the estimate and to eliminate impossible values. Delaney and Vargha (2002) used modifications of the CI for δ with Welch-like *df*s, but these modifications did not take into account specific situations in which *d* with traditional CI performed poorly. Long, et al. (2003) proposed a further adjustment on the CI to account for boundary effects on the variance of *d* due to the negative correlation between σ_d^2 and δ. Simulation studies have shown that independent *d*, when compared to the *t*-test with Welch's adjusted *df* (Welch, 1937), behaves quite well in small and moderate samples under various normal and non-normal distributions in terms of Type I error rate, power, and coverage of the CI (Feng & Cliff, 2004).

Popular statistical software packages do not include ordinal dominance analyses. Thus, the purpose of this article is to provide a Fortan program that calculates the ordinal statistic, *d*, and statistical inferences about δ , for independent groups. The program also performs Welch's *t*-test on the same data for comparison.

Du Feng is a Professor in the department of Human Development and Family Studies. Email: du.feng@ttu.edu. Norman Cliff is Professor Emeritus. Email: nrcliff5@q.com.

Methodology

Independent *d* Analysis

The calculation of independent *d* involves comparison of each of the scores in one group to each of the scores in the other group . A dominance variable d_{ij} is defined as: $d_{ij} = sign(x_i - x_j)$, where x_i represents any observation in the first group, x_j in the second. The d_{ij} simply represent the direction of differences between the x_i scores and the x_j scores: a score of +1 is assigned if $x_i > x_j$; a score of -1 is assigned if $x_i < x_j$; and a score of 0 is assigned if $x_i = x_j$. The d is an unbiased estimate of δ:

$$d = \Sigma\, \Sigma\, d_{ij} \,/\, n_1 n_2 \tag{1}$$

whereas s_d^2, the unbiased sample estimate of σ_d^2, is obtained by

$$s_d^{\,2} = \frac{n_1^{\,2}\Sigma(d_{i.} - d)^2 + n_2^{\,2}\Sigma(d_{.j} - d)^2 - \Sigma\Sigma(d_{ij} - d)^2}{n_1 n_2 (n_1 - 1)(n_2 - 1)} \tag{2}$$

where $d_{i.}$ is

$$d_{i.} = \frac{\#\left(x_i > x_j\right) - \#\left(x_i < x_j\right)}{n_1} \tag{3}$$

and similarly for $d_{.j}$. To eliminate possible negative estimate of variance, $(1 - d^2)/(n_1n_2 - 1)$ was used as the minimum allowable value for s_d^2. An asymmetric CI for δ was shown to improve the performance of d (Cliff, 1993; Feng & Cliff, 2004):

$$\delta = \frac{d - d^3 \pm Z_{\alpha/2} s_d \left(1 - 2d^2 + d^4 + Z_{\alpha/2}^{\,2} s_d^{\,2}\right)^{1/2}}{1 - d^2 + Z_{\alpha/2}^{\,2} s_d^{\,2}} \tag{4}$$

where $Z_{\alpha/2}$ is the 1- α/2 normal deviate. When d is 1.0, s_d reduces to zero, the upper bound for the CI for δ is 1.0, and the lower bound is calculated by

$$\delta = \frac{(n_{min} - Z_{\alpha/2}^{\,2})}{\left(n_{min} + Z_{\alpha/2}^{\,2}\right)} \tag{5}$$

where n_{min} is the smaller of the two sample sizes. When d equals -1.0, the solution is the negative of (5).

The Fortran Program

The Fortran program for the independent groups d analysis applies the algorithm of the above Equations (1), (2), (3), (4), and (5). The program is interactive, supplying prompts at several points. Data can be either read from a file or input from the keyboard; if input from the keyboard, data will be stored in a file. In both cases, any number of experimental variables is possible, but an analysis is conducted on only one variable at a time. After input, data are sorted within each group.

The program calculates the statistical inferences about δ, generating d and its variance, as well as the components of variance of d. The outputs include a CI for δ and the significance of d (a z-score), testing the null hypothesis. The program also calculates the dominance variable d_{ij}, and a dominance matrix for the variables analyzed is generated as a part of the outputs when the data are no more than 75 cases. Otherwise, only the statistics and their components are included in the outputs. In order to compare the d method with the classical test methods, the program also performs the classical t statistic for independent groups with Welch's adjustment of degrees of freedom. Table 1 shows an example of the output file the program generated when the sample size is 25 for both groups.

Conclusion

The ordinal method d does not involve excessive elaboration and complicated statistical analyses. Its concepts can be easily understood by non-statisticians. However, popular statistical software packages such as SAS and SPSS do not allow for ordinal dominance analyses. This Fortran program (see the appendix for source codes) for independent groups d analysis is easy to implement. Its outputs provide descriptive information, not only the null hypothesis is tested, but also a CI is provided. In addition, a dominance matrix is produced as a useful visual aid to the test. A comparison of d with Welch's t. also is provided. Furthermore, if the users have access to the IMSL library, the current source codes can be easily adapted and used in Monte Carlo studies to evaluate the performance of d in terms of Type I error rate, power, and CI coverage.

Table 1: An Example of Independent d Analysis for Two Small Samples

Ordered Scores				Dominance Diagram
Alcoholic		Non-alcoholic		
Score	$d_{i.}$	Score	$d_{.j}$	
1	–1.00	3	.92	-------------------------
4	-.72	3	.92	+++0---------------------
6	-.56	3	.92	+++++0-------------------
7	-.52	4	.88	++++++-------------------
7	-.52	5	.84	++++++-------------------
14	-.24	6	.80	+++++++++0---------------
14	-.24	12	.60	+++++++++0---------------
18	.40	12	.60	++++++++++++++++000------
19	.52	13	.60	+++++++++++++++++++------
20	.52	14	.52	+++++++++++++++++++------
21	.52	15	.44	+++++++++++++++++++------
24	.68	15	.44	+++++++++++++++++++++----
25	.68	15	.44	+++++++++++++++++++++----
26	.68	15	.44	+++++++++++++++++++++----
26	.68	15	.44	+++++++++++++++++++++----
26	.68	16	.44	+++++++++++++++++++++----
27	.72	18	.40	+++++++++++++++++++++0---
28	.84	18	.40	++++++++++++++++++++++00-
28	.84	18	.40	++++++++++++++++++++++00-
30	.92	23	.12	++++++++++++++++++++++++-
33	.92	23	.12	++++++++++++++++++++++++-
33	.92	27	-.32	++++++++++++++++++++++++-
44	1.00	28	-.44	+++++++++++++++++++++++++
45	1.00	28	-.44	+++++++++++++++++++++++++
50	1.00	43	-.76	+++++++++++++++++++++++++

Inferences About δ	
d	.389
s_d	.154
.95 confidence interval	(.063, .640)
z for d	2.530
Components of s_d^2	
$s_{d_{i.}}^2$	.394
$s_{d_{.j}}^2$	.207
$s_{d_{ij}}^2$	.831
Mean Comparisons	
t for means	2.322
Welch's df for t	44.484

References

Agresti, A. (1984). *Analysis of ordinal categorical data*. NY: Wiley.

Caruso, J. C., & Cliff, N. (1997). Empirical size, coverage, and power of confidence intervals for Spearman's rho. *Educational and Psychological Measurement*, *57*, 637-654.

Cliff, N. (1991). Ordinal methods in the study of change. In Collins, L.M. & Horn, J. (Eds.), 34-46. *Best methods for the analysis of change*. Washington, D.C.: American Psychological Association.

Cliff, N. (1993). Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological Bulletin*, *114*, 494-509.

Delaney, H. D., & Vargha, A. (2002). Comparing several robust tests of stochastic equality with ordinally scaled variables and small to moderate sized samples. *Psychological Methods*, *7*, 485-503.

Feng, D., & Cliff, N. (2004). Monte Carlo evaluation of ordinal *d* with improved confidence interval. *Journal of Modern Applied Statistical Methods*, *3*, 322-332.

Fligner, M. A., & Policello, G. E. II (1981). Robust rank procedure for the Behrens-Fisher problem. *Journal of the American Statistical Association*, 76, 162-168.

Hettmansperger, T. P. (1984). *Statistical inferences based on ranks*. NY: Wiley.

Long, J. D., Feng, D., & Cliff, N. (2003). Ordinal analysis of behavioral data. In J. Schinka, W. Velicer, and I. B. Weiner (Eds.), *Comprehensive handbook of psychology, volume two: research methods in psychology*. NY: Wiley.

Mann, H. B., & Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, *18*, 50-60.

Randles, R. H., & Wolfe, D. A. (1979). *Introduction to the Theory of Nonparametric Statistics*. NY: Wiley.

Welch, B. L. (1937). The significance of the difference between two means when the population variances are unequal. *Biometrika*, *29*, 350-362.

Appendix: Fortran Program

```
C*****************************************************************************C
C This program computes independent groups d-statistics (Cliff, 1996; Long et al.,   C
C 2003; Feng & Cliff, 2004) and provides their standard errors, confidence intervals,C
C and tests of hypotheses. The program is interactive, supplying prompts at several  C
C points. It should be noted that before doing the analyses, you should have         C
C arranged your data in the specified format.                                          C
C Data can be either read from a file or input from the keyboard. If input from the  C
C keyboard, data will be stored in a file.  Data must be entered casewise, that is,  C
C all the data for one case or person, then all for the next, etc., and we need to   C
C know the number of cases and variables.  Group membership must be entered as        C
C variable.                                                                            C
C If data are in an external file, they must be cases by variables. That is, all the C
C scores for the first case or subject, all for the second, etc. In both cases,      C
C there could be any number of experimental variables, but you can do an analysis on C
C only one variable at a time.  We need to know the number of cases, and the number  C
C of variables for each case, including the grouping variable before running the     C
C program.                                                                             C
C If the data are no more than 75 cases, a dominance matrix for the variables        C
C analyzed will be printed as part of the output. Otherwise, just the statistics and C
C their components will be included in the output.                                   C
C The program is supplied as a professional courtesy. It can be used or copied for   C
C any academic or research purpose. However, it should not be copied for any         C
C commercial purpose.  We do not know of any errors, but do not guarantee it to be   C
C errors-free.  Please understand that it was written by amateur programmers, and is C
C not intended to be of commercial quality.                                           C
C*****************************************************************************C
```

```
      INTEGER I,J,NV,NP,JQ,JC,JPLU,JG(2),NPER(2),GAP,IG,
     &        JORDER(2000,2),NDCOL(2000),NDROW(2000)
      REAL  YY,DB,SSROW,SSCOL,MINI,NUM,VARD,DEN,M1,M2,
     &      VARDROW,VARDCOL,VARDIJ,SD,UPPER,LOWER,SUM1,SUM2,MINN,
     &      SUMSQ1,SUMSQ2,VARDIFF,MDIFF,TEE,Y(2000,2),Z(2000,50)
      REAL  DEL,SQIJ,Q1,Q2,Q12
      CHARACTER*1 ANS, PLUS(3),DFILE*18,SPLU(70),SSPLU*70,
     &         STR*45, OUTFILE*8
      DATA PLUS(1),PLUS(2),PLUS(3)/'-','0','+'/
C**********************************************************************C
C Read data from a file, or input from the keyboard.                   C
C**********************************************************************C
      WRITE(*,101)
101   FORMAT('This is inddelta.f for computing d statistics.',
     & 3X,'It is copyright 1992, Norman Cliff. Comments and',
     & 1X,'suggestions are solicitted.')
      WRITE(*,102)
102   FORMAT('Type b to bypass instructions,any other letter to',
     & 1X,'see them.')
      READ(*,'(A1)') ANS
      IF((ANS.EQ.'B').OR.(ANS.EQ.'b')) GOTO 80
      WRITE(*,103)
103   FORMAT('Data can be either read form a file or input',
     & 1X,'from the keyboard. If it is in a file,it must be cases',
     & 1X,'by variabls, i.e., all the scores for the first case')
      WRITE(*,104)
104   FORMAT(' or subject, all for the second,etc. If it is not',
     & 1X,'arranged that way, type E for exit and go arrange it.')
      READ (*,'(A1)') ANS
      IF ((ANS.EQ.'E').OR.(ANS.EQ.'e')) GOTO 1500
80    WRITE(*,105)
105   FORMAT('Type f if it is in a file or k if you will enter',
     & 1X,'it from the keyboard.')
      READ(*,'(A1)')ANS
      IF((ANS.EQ.'K').OR.(ANS.EQ.'k')) THEN
        WRITE(*,111)
111     FORMAT('Data will be stored in a file. Give its full',
     & 1X,'name and extension.')
        READ(*,'(A18)') DFILE
        WRITE(*,112)
112     FORMAT('Data must be entered casewise, that is, all the',
     & 1X,'scores for one case or person, then all for the next,',1X,
     & 'etc.. And we need to know the number of cases and variables.')
        WRITE(*,113)
113     FORMAT('Group membership should be entered as a',
     & 1X,'variable.')
        WRITE(*,114)
114     FORMAT('Scores, or variables, within each case must be',
     & 1X,'separated by a comma.')
        WRITE(*,115)
115     FORMAT('No. of cases:')
        READ(*,'(I3)') NP
        WRITE(*,116)
116     FORMAT('No. of variables:')
        READ(*,'(I3)') NV
        OPEN(3,FILE=DFILE,STATUS='NEW')
        WRITE(*,117)
117     FORMAT(1X,'Enter the scores for each case, including',
     & 1X,'the grouping variable.')
        DO 1 I=1,NP
          WRITE(*,*) I
          DO 2 J=1,NV
             READ(*,*) Z(I,J)
```

```
2           CONTINUE
1         CONTINUE
         WRITE(*,118)
118      FORMAT(1X,'The scores will be printed out on the screen',
     & 1X,'for checking.')
          DO 3 I=1,NP
            WRITE(*,*) I
                WRITE(*,*) (Z(I,J),J=1,NV)
            WRITE(*,*)
3         CONTINUE
          WRITE(*,119)
119       FORMAT('If there are any corrections, type the row,',
     & 1X,'column, and the correct value. If not, type 0,0,0.')
276       READ(*,*) I,J,P
          IF(I.EQ.0) GOTO 281
          Z(I,J)=P
          WRITE(*,120)
120       FORMAT(1X,'More?  Type 0,0,0 , if not.')
          GOTO 276
281       DO 29 I=1,NP
             DO 30 J=1,NV
                WRITE(1,*) Z(I,J)
30           CONTINUE
29        CONTINUE
          CLOSE (3,STATUS='KEEP')
        ELSE
            IF((ANS.NE.'F').AND.(ANS.NE.'f')) THEN
                  GOTO 80
            ELSE
               WRITE(*,106)
106            FORMAT('Type name of file, including extention,',
     & 1X,'also path if not in this directory.')
                WRITE(*,107)
107             FORMAT('filename')
                READ(*,'(A18)') DFILE
                WRITE(*,108)
108             FORMAT('How many variables per case?')
                READ(*,'(I2)') NV
                WRITE(*,109)
109             FORMAT('How many cases?')
                READ(*,'(I3)') NP
               OPEN(4,FILE=DFILE,STATUS='OLD')
               DO 31 I=1,NP
                  READ(4,*) (Z(I,J), J=1,NV)
31             CONTINUE
               CLOSE(4,STATUS='KEEP')
            ENDIF
         ENDIF
282    WRITE(*,122)
122    FORMAT('Which variable no. is the grouping variable?')
       READ(*,'(I1)') JC
       WRITE(*,123)
123    FORMAT('Which variable no. is the experimental?')
       READ(*,'(I1)') JQ
       WRITE(*,124)
124    FORMAT('Which are two values of the grouping variable',1X,
     & 'designate the groups to be compared?(e.g.:1 and 2)')
       WRITE(*,125)
125    FORMAT(1X,'  First group: ')
       READ(*,'(I2)') JG(1)
       WRITE(*,126)
126    FORMAT(1X,'  Second group: ')
       READ(*,'(I2)') JG(2)
```

```
      NPER(1) = 1
      NPER(2) = 1
      WRITE(*,226)
226   FORMAT(1X,' Name of the output file is: ')
      READ(*,'(A9)') OUTFILE
      OPEN(8,FILE=OUTFILE)
C***********************************************************************C
C   Sort data.                                                          C
C***********************************************************************C
      DO 4  I=1,NP
       IF(Z(I,JC).EQ.JG(1)) THEN
         Y(NPER(1),1) = Z(I,JQ)
         JORDER(NPER(1),1) = NPER(1)
         NPER(1) = NPER(1)+1
       ELSE IF (Z(I,JC).EQ.JG(2)) THEN
         Y(NPER(2),2) = Z(I,JQ)
         JORDER(NPER(2),2) = NPER(2)
         NPER(2) = NPER(2)+1
       ELSE
       ENDIF
4     CONTINUE
       NPER(1)=NPER(1)-1
       NPER(2)=NPER(2)-1
       WRITE(*,127) NPER(1),NPER(2)
127    FORMAT(1X,2I4)
      DO 5  IG=1,2
        DO 6 K=4,1,-1
         GAP=2**K-K
           DO 7 I=GAP,NPER(IG)
            XX=Y(I,IG)
            YY=JORDER(I,IG)
            J=I-GAP
430         IF((J.LE.0).OR.(XX.GE.Y(J,IG))) GOTO 450
            Y(J+GAP,IG)=Y(J,IG)
            JORDER(J+GAP,IG)=JORDER(J,IG)
            J=J-GAP
            GOTO 430
450         Y(J+GAP,IG)=XX
            JORDER(J+GAP,IG)=YY
7          CONTINUE
6        CONTINUE
5      CONTINUE
C***********************************************************************C
C     Calculate dominance matrix (and print the matrix for small data set).   C
C***********************************************************************C
         SQIJ = 0.0
         DEL= 0.0
       WRITE(8,131)
131    FORMAT(1X,'This is an independent data analysis using',1X,
     & 'inddelta.f.')
       WRITE(8,*)
       WRITE(8,132) DFILE
132    FORMAT(1X,'The data are from ',A18)
       WRITE(8,*)
       WRITE(8,133) NV-1
133    FORMAT(1X,'There are ',I3,' experimental variable(s).')
       WRITE(8,*)
       WRITE(8,134) JC
134    FORMAT(1X,'The grouping variable is ',I3)
       WRITE(8,135) JQ
135    FORMAT(1X,'The experimental variable is ',I3)
       WRITE(8,*)
       DO 999 I = 1,NPER(1)
```

```
                 NDROW(I) = 0
999     CONTINUE
        DO 998 I = 1,NPER(2)
                 NDCOL(I) = 0
998          CONTINUE
         IF(NP.LE.75) THEN
           WRITE(8,137) JG(1),JG(2)
137        FORMAT(1X,'Dominance matrix for group',I3,' vs. group',I3)
           WRITE(8,*)
           WRITE(8,138) JG(1),JG(2)
138        FORMAT(1X,'A + INDICATES ',I3,' HIGHER THAN',I3)
           WRITE(8,*)
           DO 9 I=1,NPER(1)
                SSPLU = ' '
             DO 10 J=1,NPER(2)
                   IF(Y(I,1).GT.Y(J,2)) THEN
                      IWON=1
                   ELSE IF(Y(I,1).LT.Y(J,2)) THEN
                      IWON=-1
                   ELSE
                      IWON=0
                   ENDIF
                 DEL = DEL +IWON
                 SQIJ = SQIJ+IWON*IWON
                 NDROW(I) = NDROW(I)+IWON
                 NDCOL(J) = NDCOL(J)+IWON
                   JPLU = IWON + 2
                   SPLU(J) = PLUS(JPLU)
                   SSPLU = SSPLU(1:J)//SPLU(J)
10           CONTINUE
                 WRITE(8,139) SSPLU
139              FORMAT(1X,A72)
9          CONTINUE
           WRITE(8,*)
           WRITE(8,*)
           WRITE(8,*)
         ELSE
           DO 11 I=1,NPER(1)
             DO 12 J=1,NPER(2)
                   IF(Y(I,1).GT.Y(J,2)) THEN
                      IWON=1
                   ELSE IF(Y(I,1).LT.Y(J,2)) THEN
                      IWON=-1
                   ELSE
                      IWON=0
                   ENDIF
                 DEL = DEL +IWON
                 SQIJ = SQIJ+IWON*IWON
                 NDROW(I) = NDROW(I)+IWON
                 NDCOL(J) = NDCOL(J)+IWON
12           CONTINUE
11          CONTINUE
         ENDIF
C**********************************************************************C
C     Calculate d and variance of d.                                   C
C**********************************************************************C
        DB = DEL/(NPER(1)*NPER(2))
        WRITE(8,*)
        WRITE(8,140)
140     FORMAT(1X,'***',2X,'d and its variance',2X,'***')
        WRITE(8,141) JG(1),JG(2),DB
141     FORMAT(1X,'d for ',I3,' vs. ',I3,27X,'  =  ',F6.3)
```

```
C*********************************************************************C
C       This part is for calculations of variance of d.               C
C*********************************************************************C
         SSROW=0.0
         SSCOL=0.0
         DO 14 I=1,NPER(1)
         SSROW = SSROW + NDROW(I)**2
14       CONTINUE
         DO 15 I=1,NPER(2)
         SSCOL = SSCOL + NDCOL(I)**2
15       CONTINUE
         MINI=(SQIJ/(NPER(1)*NPER(2))-DB**2)
     &       /(NPER(1)*NPER(2)-1)
         NUM=SSROW-NPER(2)*DEL*DB + SSCOL - NPER(1)*DEL*DB
     &       -SQIJ + DEL*DB
         DEN = NPER(1)*NPER(2)*(NPER(1) - 1)*(NPER(2)-1)
         VARD = NUM/DEN
         IF (VARD.LE. MINI) THEN
            VARD = MINI
            WRITE(8,142)
142         FORMAT(1X,'variance = minimum.Interpret with caution.')
         ELSE
         ENDIF
         STR='variance for d'
         WRITE(8,143) STR,VARD
143      FORMAT(1X,A45,'  =  ',F7.4)
         VARDROW = (SSROW - NPER(2)*DEL*DB)
     &           /(NPER(2)**2*(NPER(1) - 1))
         VARDCOL = (SSCOL - NPER(1)*DEL*DB)
     &            /(NPER(1)**2*(NPER(2) - 1))
         VARDIJ = (SQIJ - DEL*DB)/(NPER(1)*NPER(2) - 1)
         WRITE(8,*)
         WRITE(8,144)
144      FORMAT(10X,'***  Components of the variance of d :  ***')
         STR='row di variance '
         WRITE(8,145) STR,VARDROW
145      FORMAT(1X,A45,'  =  ',F7.4)
         STR='column di variance '
         WRITE(8,146) STR,VARDCOL
146      FORMAT(1X,A45,'  =  ',F7.4)
         STR='variance of dij'
         WRITE(8,147) STR,VARDIJ
147      FORMAT(1X,A45,'  =  ',F7.4)
         SD = SQRT(VARD)
C*********************************************************************C
C   Calculate the asymmetric 95% confidence interval for delta,       C
C   with further agjustment on C.I. when d = 1.0 or d = -1.0.         C
C*********************************************************************C
         IF (NPER(1).LE.NPER(2)) THEN
            MINN = NPER(1)
         ELSE
            MINN = NPER(2)
         ENDIF
         IF (DB.EQ.1.0) THEN
            UPPER = 1.0
            LOWER = (MINN - 1.96**2)
     &             /(MINN + 1.96**2)
         ELSE IF (DB.EQ.(-1.0)) THEN
            LOWER = -1.0
            UPPER = -(MINN - 1.96**2)
     &              /(MINN + 1.96**2)
```

```
         ELSE
            UPPER = (DB-DB**3 + 1.96*SD*SQRT(DB**4 - 2*DB**2 + 1
     &+ 1.96*1.96*VARD)) / (1-DB**2 + 1.96*1.96*VARD)
             IF (UPPER.GT.1) UPPER = 1
            LOWER = (DB-DB**3 - 1.96*SD*SQRT(DB**4 - 2*DB**2 + 1
     &+ 1.96*1.96*VARD)) / (1-DB**2 + 1.96*1.96*VARD)
             IF (LOWER.LT.-1) LOWER = -1
         ENDIF
         WRITE(8,148)
148      FORMAT(10X,'**  Inference :  **')
         STR='approximate .95 Confidence limits for d '
         WRITE(8,149) STR
149      FORMAT(1X,A40)
         WRITE(8,*)
         IF(UPPER.GT.1) UPPER = 1
         IF(LOWER.LT.-1) LOWER = -1
         WRITE(8,150) LOWER,UPPER
150      FORMAT(20X,F6.3,' to ',F6.3)
         WRITE(8,*)
         STR='significance of d :'
         WRITE(8,151) STR,DB/SD
151      FORMAT(1X,A45,' z   =  ',F7.4)
         WRITE(8,*)
C*******************************************************************************C
C        This short section computes the ordinary unpooled t-test              C
C        with Welch's adjusted df.                                             C
C*******************************************************************************C
         SUM1 = 0.0
         SUM2 = 0.0
         SUMSQ1 = 0.0
         SUMSQ2 = 0.0
         DO 20 I = 1,NPER(1)
            SUM1 = SUM1 + Y(I,1)
20       CONTINUE
         M1 = SUM1/NPER(1)
         DO 21 I =1,NPER(1)
            SUMSQ1 = SUMSQ1 + (Y(I,1) - M1)**2
21       CONTINUE
         DO 22 I =1,NPER(2)
            SUM2 = SUM2 + Y(I,2)
22       CONTINUE
         M2 = SUM2/NPER(2)
         DO 23 I =1,NPER(2)
            SUMSQ2 = SUMSQ2 + (Y(I,2)-M2)**2
23       CONTINUE
         Q1 = SUMSQ1/(NPER(1)*(NPER(1)-1))
         Q2 = SUMSQ2/(NPER(2)*(NPER(2)-1))
         VARDIFF = Q1 + Q2
         MDIFF = M1 - M2
         TEE = MDIFF/SQRT(VARDIFF)
         WRITE(8,152)
152      FORMAT(6X,'***  Independent t-test with unpooled variance :',
     & 1X,' *** ')
         STR='mean difference'
         WRITE(8,153) STR,MDIFF
153      FORMAT(1X, A45,'  =  ',F7.4)
         STR='standard deviations:'
         WRITE(8,154) STR,SQRT((SUMSQ1/(NPER(1) - 1))),
     &             SQRT((SUMSQ2/(NPER(2)-1)))
154      FORMAT(1X,A47,'(1) ',F7.4,'  (2)   ',F7.4)
         STR='standard error of mean difference'
```

```
      WRITE(8,155) STR,SQRT(VARDIFF)
155   FORMAT(1X,A45,'  =  ',F7.4)
      Q12=(Q1+Q2)**2/(Q1**2/(NPER(1)-1)+Q2**2/(NPER(2)-1))
      WRITE(8,156) TEE,Q12
156   FORMAT(9X,'t  =  ',F8.4,9X,'adjusted df  =   ',F8.4)
      WRITE(8,*)
      WRITE(*,157)
157   FORMAT('Do you want the data to be printed on the',1X,
     &       'printer, y/n?')
      READ(*,'(A1)') ANS
      IF((ANS.EQ.'Y').OR.(ANS.EQ.'y')) THEN
          WRITE(8,*)
          WRITE(8,158)
158       FORMAT(10X,'***  Ordered data for this variable :  ***')
          WRITE(8,159)
159       FORMAT(1X,'ORDER',5X,'SUBJ.',5X,'SCORE',5X,'ROWDOM')
          WRITE(8,160) JG(1)
160       FORMAT(1X,'Group ',I3)
          DO 25 I=1,NPER(1)
              WRITE(8,161) I,JORDER(I,1),Y(I,1),NDROW(I)
161           FORMAT(1X,I5,5X,I5,5X,F6.3,5X,I3)
25        CONTINUE
          WRITE(8,162) JG(2)
162       FORMAT(1X,'Group ',I3)
          DO 26 I=1,NPER(2)
              WRITE(8,163) I,JORDER(I,2),Y(I,2),NDCOL(I)
163           FORMAT(1X,I5,5X,I5,5X,F6.3,5X,I3)
26        CONTINUE
      ELSE
      ENDIF
C*****************************************************************************C
      WRITE(8,*)
      WRITE(8,*)
      WRITE(8,*)
      WRITE(*,164)
164   FORMAT('Do you want to do another analysis, y or n?')
      READ(*,'(A1)') ANS
      IF (ANS.EQ.'Y'.OR.ANS.EQ.'y')  GOTO 282
1500  CLOSE(8,STATUS="KEEP")
      END
```