

TWO MACROS FOR CALCULATING OFTEN USED STATISTICS OF SYMMETRY AND AGREEMENT IN THE BIOPHARMACEUTICAL INDUSTRY

Jeff Heckathorn and Mark F. Munsell
Applied Logic Associates, Inc.

ABSTRACT

When interest lies in measuring symmetry between two categorical variables, statistics such as those proposed by Stuart [1955], Maxwell [1970] and McNemar [1947] are often employed. When interest lies in determining the extent of agreement (or correlation) between two or more raters or measuring devices, the concordance rate or the kappa statistic proposed by Cohen [1960] may be employed. When interest lies in describing the relationship between a screening test (or symptom) and the disease to which it relates, statistics such as sensitivity, specificity, positive predictive value (PV+), and negative predictive value (PV-) are generally employed. This poster presents two (2) macros which are actively used in the biopharmaceutical industry to calculate and present the above statistics with appropriate confidence intervals (binomial [1973] or normal) and p-values.

INTRODUCTION

Often in the field of biopharmaceutical consulting there is a need to compare a newly developed diagnostic test with a similar test that is generally accepted as a standard. Another situation which often arises involves the comparison of two "raters" of an experimental unit, such as the disease status of a patient, to determine the degree of agreement between the two raters. A third situation which is often encountered is the tracking of an experimental unit over time and evaluating changes which may occur with time. Many of the statistics described below can be used in more than one of the sample scenarios to provide additional insight or explanation.

With the first scenario described above, common statistics which are desired are the concordance rate (percent of time that the two diagnostic tests agree), sensitivity (probability of testing positive given that the disease is present), specificity (probability of testing negative given that the disease is not present), predictive value positive (probability that the disease is present given that the test is positive), and predictive value negative (probability that the disease is not present given that the test is negative). Along with the point estimates of these several probabilities of interest it is useful to have confidence intervals.

The second scenario mentioned above involves the grading into discrete categories the disease status of patients by two independent raters. The degree of agreement between these two raters is of interest, since if two raters cannot agree on a diagnosis the course of treatment may be affected. A kappa statistic and associated confidence interval provide a measure of agreement between the two raters.

The third situation discussed above may involve the measurement of a particular blood level at a baseline period and then again at a later time. The particular parameter measured can be classified as low, normal, or high each time it is measured. What is of interest here is whether there has been a change in classification for the cohort of patients on which the blood level is measured. For instance, have patients who were measured as normal at baseline measured low or high at the subsequent visit? To get an idea of whether such changes have taken place, the McNemar or Stuart-Maxwell statistic can be used to provide a measure of symmetry.

DISCUSSION OF MACROS

Macro 1: The RUNTAB Macro

Macro 1 can be used for 2 x 2 tables to compare a standard test (process) with a newly developed test (process). The main activities of Macro 1 are to produce a 2 x 2 table from the data and calculate sensitivity, specificity, predictive value positive, and predictive value negative with corresponding 95% binomial confidence intervals (large sample approximations). The main activities of Macro 1 will be briefly explained in the following steps (which correspond to the steps listed in the program itself).

****** One important note on the temporary dataset T1 which is used in this macro is that there is a variable called DUMM which must be present and whose value is 1 for each record. We use DUMM in most of our Proc Tabulates to designate actual observations as opposed to generated dummy records used to produce full tables. DUMM is also used in subsequent data steps with a BY statement to utilize the FIRST.variable or LAST.variable options.

Step 1: Macro 1 begins with a Proc Tabulate to generate the 2 x 2 table with row percentages. Next, Proc Freq is used to generate the required cell counts and output these records to the FREQDAT1 dataset. Care should be taken to consider the order of the variables so that the calculated statistics will be correct. For this macro, the variables take on values of 0 or 1, where 0 = normal (no disease) and 1 = abnormal (disease). Finally, in the FREQDAT data step, numerous arrays are declared, the counts are put into an appropriate array, the elements of the count array are retained since only the last record is used for the remainder of the data step, missing values for the elements of the count array are set to zero (0), and we proceed to Step 2.

Step 2: The TOT array is filled with the marginal counts. If an element of the TOT array equals zero (0), then its value is set to missing (.) to avoid division by zero (0).

Step 3: Calculate statistics, set $(1-\alpha/2)$ for use with FINV function later on, and fill in array T with marginal counts and array B with cell counts to facilitate programming.

Step 4: Calculate the numerator and denominator degrees of freedom which are then used in the calculation of the lower and upper confidence limits for the above statistics.

Step 5: Use FORMDLIM= option to eliminate page break so that Proc Print output appears on the same page as the table, print out statistics and confidence limits, and end Macro 1.

Step 6: A sample call to Macro 1 which generates Table I.

Macro 2: The RUNPROC Macro

Macro 2 can be used for any k x k table to investigate changes in categorical classifications which may occur over time and the degree of agreement. The main activities of Macro 2 are the tabulation of baseline (some timepoint) by subsequent visit (later timepoint) categories for specified variables and the calculation and output of a Stuart-Maxwell or McNemar's test statistic, the concordance rate, the kappa statistic, and 95% binomial confidence intervals (large sample approximations) or p-values where appropriate. The main activities of Macro 2 will be briefly explained in the following steps (which correspond to the steps listed in the program itself).

****** The same note for the temporary dataset T1 as discussed previously also applies to the temporary datasets T3 and T4 used in Macro 2.

Step 1: Macro 2 will work for any size dimension DIM, but the DIM must be passed to the macro. DIM should represent the maximum number of categories that each variable is capable of having. Macro 2 begins by declaring PRINT a global variable and using the DIM variable to calculate the DIMSQ and DIMPL variables for later use. Next, Proc Tabulate is used to generate the appropriate

k x k table with row percentages and uses the DUMM variable to indicate true observations. Temporary dataset, TEMP1, determines whether the dataset to be analyzed has observations and sets a FLAG. The remainder of the macro is run only if observations exist. Proc Freq is used to generate the required cell counts and output these records to the FREQDAT1 dataset.

Step 2: Call to the STU_MAX macro which starts off sorting FREQDAT1.

Step 3: Intermediate data step used for analysis variables whose category values start with zero (0). Since array subscripts must be greater than zero (0), this data step increments each value by 1 for the analysis. Begin INVERT data step by declaring a two-dimensional array A to hold cell counts.

Step 4: Initialize array A elements to missing.

Step 5: Fill in array A with cell counts from earlier Proc Freq, replace missing values with zero (0), retain array A elements since only the last record will be kept, begin INVERT1 data step by declaring array A and arrays R & C (for column totals and row totals) and initializing the CNT variable to zero (0).

Step 6: Initialize arrays R & C to zero (0).

Step 7: Calculate row and column totals.

Step 8: Begin matrix (array) reduction section of STU_MAX macro to prevent singularity when zero rows or columns exist.

Step 9: If corresponding rows and columns are all zeros (0), then set their element values to missing (.) and increment CNT to count each time this happens.

Step 10: Matrix (array A) reduction to remove corresponding rows and columns of zeros, since corresponding zero rows and columns add nothing to subsequent calculations.

Step 11: Create a macro variable NEW to hold the

current array A dimension.

Step 12: Set the value of the macro variable PRINT to one (1) or two (2), depending on whether the value of NEW is greater than two (2) or equal to two (2). PRINT determines which Proc Print will be used, the one for Stuart-Maxwell or the one for McNemar's.

Step 13: Create macro variables DIM2, DIMSQ2, and DIMPL2 which correspond to new array A dimension for later matrix (array) operations.

Step 14: If NEW = 2, then the STU_MAX macro will calculate McNemar's test statistic and p-value. Begin INVERT2 data step, calculate DENOM (total discordant pairs), and assign PRINT a value of missing (.) if there are no discordant pairs.

Step 15: Use normal approximation to calculate McNemar's test statistic if the number of discordant pairs exceeds 20.

Step 16: Use exact binomial calculation of McNemar's test statistic if the number of discordant pairs is less than or equal to 20. Flag any p-value that is less than 0.05 with an asterisk (*). Begin INVERT2 data step for when NEW is greater than 2 by declaring arrays NN, D, V, & A and getting data ready to invert the covariance matrix as part of the process to calculate the Stuart-Maxwell statistic.

Step 17: Invert the covariance matrix (array V) with the sweep operator.

Step 18: Calculate the Stuart-Maxwell statistic (a Chi-Square statistic) and the associated p-value, flag any p-value that is less than 0.05, and end the STU_MAX macro.

Step 19: Call the RUNPRINT macro. Pick the appropriate Proc Print to use according to the value of the PRINT macro variable. End the RUNPRINT macro.

Step 20: Begin the FREQDAT data step to be used with the RUNCONC and RUNKAP macros. If any category value is missing, set that

record's COUNT and PERCENT variables to zero (0). Sort FREQDAT.

Step 21: Call the RUNCONC macro (which is similar to **MACRO 1**). Begin the CONCORD data step by initializing the TOTAL (total number of records in dataset) and AGREE (total number of records that are in agreement at baseline and subsequent visits) variables to zero (0). Accumulate TOTAL. Accumulate AGREE. Retain TOTAL and AGREE since only the last record will be kept. Set the PERCENT (concordance rate), PROB_L (lower confidence limit), and PROB_U (upper confidence limit) to missing (.) if TOTAL equals zero (0) to prevent division by zero (0). Calculate PERCENT and set for use with FINV function if TOTAL is not equal to zero (0).

Step 22: Continuing when TOTAL is not equal to zero (0), calculate upper and lower 95% binomial confidence intervals (large sample approximations) as described in **MACRO 1**. Proc Print used to output concordance statistics beneath table from Proc Tabulate. End the RUNCONC macro.

Step 23: Call the RUNKAP macro. Begin the KAPPA data step by declaring arrays PCT (cell percent), PCTCOL (column marginal percent), and PCTROW (row marginal percent). Initialize the elements of array PCT and CNT (a counter) to zero (0). Assign values to elements of array PCT from the output of Proc Freq in Step 20. Accumulate the total number of records in array PCT in variable N. Retain elements of array PCT and N since only the last record will be kept. Initialize the elements of arrays PCTCOL and PCTROW to zero (0).

Step 24: Calculate column marginal percents.

Step 25: Initialize counter CNT2 to zero (0) and calculate row marginal percents.

Step 26: Initialize P0, the numerator of the kappa statistic, to zero (0). Calculate P0.

Step 27: Initialize PE, the denominator of the kappa statistic, to zero (0). Calculate PE. If PE equals one (1), then set KAPPA (the kappa statistic) and SE (standard error of kappa) to missing (.) to prevent division by zero (0). If PE does not equal one (1), calculate KAPPA, initialize SUM_TERM (part of SE equation) to zero (0), calculate SUM_TERM, SE, UPPER (upper 95% confidence limit), and LOWER (lower 95% confidence limit).

Step 28: Test the null hypothesis, $H_0: KAPPA = 0$. Calculate the z statistic and corresponding p-value. Proc Print used to output the kappa statistics beneath the concordance statistics. End RUNKAP macro. End Macro 2.

Step 29: Sample macro calls to Macro 2 which generate Tables II.A - II.C

REFERENCES

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educ. Psychol. Meas.*, 20, 37-46.

Hollander, Myles and Wolfe, Douglas A. (1973). Nonparametric Statistical Methods. New York: John Wiley & Sons, 23-24.

Maxwell, A. E. (1970). Comparing the classification of subjects by two independent judges. *Brit. J. Psychiatry*, 116, 651-655.

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12, 153-157.

Stuart, A. (1955). A test for homogeneity of the marginal distribution in a two-way classification. *Biometrika*, 42, 412-416.

```

*****
*** MACRO 1: A MACRO FOR 2x2 TABLES WHICH TABULATES VARIABLES ***
*** OF INTEREST AND CALCULATES SENSITIVITY, SPECIFICITY, ***
*** PREDICTIVE VALUE POSITIVE, AND PREDICTIVE VALUE NEGATIVE ***
*** WITH 95% BINOMIAL CONFIDENCE INTERVALS. ***
*****

%MACRO RUNTAB (VAR1, VAR2, VARNAME);

*****
*** STEP 1 ***
*****

PROC TABULATE DATA = T1;
  CLASS &VAR1 &VAR2;
  TABLE &VAR1 ALL, (&VAR2 ALL)*
           (N='N'*F=6, PCTN=&VAR2 ALL=>'X'*F=6.2)
           / MISSTEXT = '0' RTS = 20;
  KEYLABEL ALL = 'TOTAL';
  LABEL &VAR1 = 'STANDARD TREATMENT'
        &VAR2 = 'EXPERIMENTAL &VARNAME';
  TITLE1 'TABLE 1';
  TITLE2 'TABULATION AND ANALYSIS OF';
  TITLE3 'STANDARD TREATMENT VS. EXPERIMENTAL &VARNAME';
  TITLE4 'FOR VARIABLE OF INTEREST';
RUN;

PROC FREQ DATA = T1;
  BY DUMM;
  TABLES &VAR1*&VAR2 / NOPRINT OUT = FREQDAT1;
RUN;

DATA FREQDAT;
  SET FREQDAT1;
  BY DUMM;

  ARRAY CNT(4) CNT1-CNT4;
  ARRAY TOT(4) TOT1-TOT4;
  ARRAY T(4) T1-T4;
  ARRAY B(4) B1-B4;
  ARRAY D_DFL(4) D_DFL1-D_DFL4;
  ARRAY D_DFU(4) D_DFU1-D_DFU4;
  ARRAY N_DFL(4) N_DFL1-N_DFL4;
  ARRAY N_DFU(4) N_DFU1-N_DFU4;
  ARRAY PROBL(4) PROBL1-PROBL4;
  ARRAY PROBFU(4) PROBFU1-PROBFU4;
  ARRAY PROB_L(4) PROB_L1-PROB_L4;
  ARRAY PROB_U(4) PROB_U1-PROB_U4;

  IF &VAR1 = 1 AND &VAR2 = 1 THEN CNT1 = COUNT;
  IF &VAR1 = 2 AND &VAR2 = 1 THEN CNT2 = COUNT;
  IF &VAR1 = 1 AND &VAR2 = 2 THEN CNT3 = COUNT;
  IF &VAR1 = 2 AND &VAR2 = 2 THEN CNT4 = COUNT;

  RETAIN CNT1-CNT4;

  IF LAST.DUMM;

  DO I = 1 TO 4;
    IF CNT(I) = . THEN CNT(I) = 0;
  END;

```

```

*****
*** STEP 2: MARGINAL COUNTS ***
*****
  TOT1 = SUM(CNT1, CNT2);
  TOT2 = SUM(CNT3, CNT4);
  TOT3 = SUM(CNT1, CNT3);
  TOT4 = SUM(CNT2, CNT4);

  DO I = 1 TO 4;
    IF TOT(I) = 0 THEN TOT(I) = .;
  END;

*****
*** STEP 3: CALCULATION OF STATISTICS ***
*****
  SPECIF = 100 * CNT1 / TOT3;
  SENSIT = 100 * CNT4 / TOT4;
  PREDVPOS = 100 * CNT4 / TOT2;
  PREDVNEG = 100 * CNT1 / TOT1;

  PROB = 1 - 0.05/2;

  T1 = TOT3; T2 = TOT4; T3 = TOT2; T4 = TOT1;
  B1 = CNT1; B2 = CNT4; B3 = CNT3; B4 = CNT2;

  *****
  *** STEP 4 ***
  *****

  *****
  *** NUMERATOR DEGREES OF FREEDOM USED IN ***
  *** LOWER CONFIDENCE INTERVAL CALCULATION ***
  *****
  DO I = 1 TO 4;
    N_DFL(I) = 2*(T(I) - B(I) + 1);
  END;

  *****
  *** DENOMINATOR DEGREES OF FREEDOM USED IN ***
  *** LOWER CONFIDENCE INTERVAL CALCULATION ***
  *****
  DO I = 1 TO 4;
    D_DFL(I) = 2*B(I);
  END;

  *****
  *** CALCULATION OF LOWER CONFIDENCE LIMITS ***
  *****
  DO I = 1 TO 4;
    IF D_DFL(I) = 0 AND T(I) NE . THEN PROB_L(I) = 0;
  ELSE DO;
    PROBL(I) = FINV( PROB , N_DFL(I), D_DFL(I));
    PROB_L(I) = 100 * B(I) /
                (B(I) + (T(I) - B(I) + 1)*PROBL(I));
  END;
  END;

  *****
  *** NUMERATOR DEGREES OF FREEDOM USED IN ***
  *** UPPER CONFIDENCE INTERVAL CALCULATION ***
  *****
  DO I = 1 TO 4;
    N_DFU(I) = 2*(B(I) + 1);
  END;

```

```

*****
*** DENOMINATOR DEGREES OF FREEDOM USED IN ***
*** UPPER CONFIDENCE INTERVAL CALCULATION ***
*****
  DO I = 1 TO 4;
    D_DFU(I) = 2*(T(I) - B(I));
  END;

  *****
  *** CALCULATION OF UPPER CONFIDENCE LIMITS ***
  *****
  DO I = 1 TO 4;
    IF D_DFU(I) = 0 AND T(I) NE . THEN PROB_U(I) = 100;
  ELSE DO;
    PROBFU(I) = FINV(PROB, N_DFU(I), D_DFU(I));
    PROB_U(I) = 100 * (1 - ((T(I) - B(I)) /
                          (T(I) - B(I)) + (B(I) + 1)*PROBFU(I)))));
  END;
  END;
RUN;

  *****
  *** STEP 5 ***
  *****

  OPTIONS FORMLIN = ' ';

  PROC PRINT DATA = FREQDAT NOOBS LABEL SPLIT = '/';
    VAR SENSIT PROB_L2 PROB_U2 SPECIF PROB_L1 PROB_U1;
    FORMAT SENSIT PROB_L2 PROB_U2 SPECIF PROB_L1 PROB_U1 B.3;
    LABEL SPECIF = 'SPECIFICITY/(%)'
           SENSIT = 'SENSITIVITY/(%)'
           PROB_L1 = 'LOWER 95%/BINOMIAL C.I.'
           PROB_U1 = 'UPPER 95%/BINOMIAL C.I.'
           PROB_L2 = 'LOWER 95%/BINOMIAL C.I.'
           PROB_U2 = 'UPPER 95%/BINOMIAL C.I.';
    TITLE;
  RUN;

  PROC PRINT DATA = FREQDAT NOOBS LABEL SPLIT = '/';
    VAR PREDVPOS PROB_L3 PROB_U3 PREDVNEG PROB_L4 PROB_U4;
    FORMAT PREDVPOS PROB_L3 PROB_U3 PREDVNEG PROB_L4 PROB_U4 B.3;
    LABEL PREDVPOS = 'PREDICTIVE VALUE/POSITIVE/PV(+)'
           PREDVNEG = 'PREDICTIVE VALUE/NEGATIVE/PV(-)'
           PROB_L3 = 'LOWER 95%/BINOMIAL C.I.'
           PROB_U3 = 'UPPER 95%/BINOMIAL C.I.'
           PROB_L4 = 'LOWER 95%/BINOMIAL C.I.'
           PROB_U4 = 'UPPER 95%/BINOMIAL C.I.';
    TITLE;
  RUN;

  OPTIONS FORMLIN = ' ';

  %MEND RUNTAB;

  *****
  *** STEP 6 ***
  *****

  *****
  *** A SAMPLE CALL TO THE ABOVE MACRO ***
  *****

  %RUNTAB (STANDARD, TREAT, TREATMENT 1);

```

TABLE I
TABULATION AND ANALYSIS OF
STANDARD TREATMENT VS. EXPERIMENTAL TREATMENT 1
FOR VARIABLE OF INTEREST

	EXPERIMENTAL TREATMENT 1					
	NORMAL		ABNORMAL		TOTAL	
	N	%	N	%	N	%
STANDARD TREATMENT						
NORMAL	24	64.86	13	35.14	37	100.00
ABNORMAL	56	35.90	100	64.10	156	100.00
TOTAL	80	41.45	113	58.55	193	100.00

SENSITIVITY (%)	LOWER 95% BINOMIAL C.I.	UPPER 95% BINOMIAL C.I.	SPECIFICITY (%)	LOWER 95% BINOMIAL C.I.	UPPER 95% BINOMIAL C.I.
64.103	56.042	71.616	64.865	47.461	79.790

PREDICTIVE VALUE POSITIVE PV(+)	LOWER 95% BINOMIAL C.I.	UPPER 95% BINOMIAL C.I.	PREDICTIVE VALUE NEGATIVE PV(-)	LOWER 95% BINOMIAL C.I.	UPPER 95% BINOMIAL C.I.
88.496	81.131	93.730	30.000	20.257	41.276

```

*****
*** MACRO 2: THE MAIN MACRO WHICH DIRECTS THE TABULATION OF BASE-
*** LINE BY SUBSEQUENT VISIT CATEGORIES FOR SPECIFIED VARIABLES
*** AND CALCULATES AND PRINTS OUT STUART-MAXWELL OR MCNEMAR'S TEST
*** STATISTIC, CONCORDANCE RATE, AND KAPPA STATISTIC WITH 95%
*** CONFIDENCE INTERVALS AND P-VALUES WHERE APPROPRIATE.
*****

```

```

%MACRO RUNPROC(TABNUM, NUM, TITLE, FMT, DIM) ;

```

```

*****
*** STEP 1 ***
*****

```

```

%GLOBAL PRINT ;

```

```

%LET DIMSQ = %EVAL(&DIM**2) ;
%LET DIMPL = %EVAL(&DIM**2) ;

```

```

PROC TABULATE DATA = T4 ;

```

```

WHERE BASE&NUM NE . AND PARAM&NUM NE . ;
CLASS BASE&NUM PARAM&NUM ;
VAR DUMM ;

```

```

TABLES BASE&NUM ALL, PARAM&NUM'DUMM' *
(SUM*F=8.0 PCTSUM=PARAM&NUM**F=8.2)
/ MISSTEXT = '0' RTS = 20 ;

```

```

KEYLABEL SUM = 'N'
ALL = 'TOTAL'
PCTSUM = 'X' ;

```

```

FORMAT BASE&NUM PARAM&NUM &FMT ;
LABEL

```

```

PARAM&NUM = 'SUBSEQUENT VISIT'
BASE&NUM = 'BASELINE VISIT' ;

```

```

TITLE1 'Table II.&TABNUM' ;
TITLE2 'Clinical Assessment' ;
TITLE3 'Tabulation of Baseline Visit by Subsequent Visit' ;
TITLES 'FOR &TITLE' ;

```

```

RUN ;

```

```

DATA TEMP1 ;

```

```

SET T4 ;
WHERE BASE&NUM NE . AND PARAM&NUM NE . ;
IF _N_ = 1 THEN CALL SYMPUT('FLG', .) ;
IF DUMM = 1 THEN CALL SYMPUT('FLG', 1) ;

```

```

RUN ;

```

```

%IF &FLG = . %THEN %LET FLAG = . ;
%ELSE %LET FLAG = 1 ;

```

```

%IF &FLAG = 1 %THEN %DO ;

```

```

PROC FREQ DATA = T3 ;

```

```

BY DUMM ;
TABLES BASE&NUM * PARAM&NUM / OUT=FREQDAT1 NOPRINT ;
RUN ;

```

```

*****

```

```

*** STEP 2 ***

```

```

*****

```

```

%STU_MAX (&NUM, &DIM, &DIMSQ, &DIMPL) ;

```

```

*****

```

```

*** STEP 19 ***

```

```

*****

```

```

%RUNPRINT ;

```

```

*****

```

```

*** STEP 20 ***

```

```

*****

```

```

DATA FREQDAT ;
SET FREQDAT1 ;

```

```

IF BASE&NUM = . OR PARAM&NUM = . THEN DO ;

```

```

COUNT = 0 ;

```

```

PERCENT = 0 ;

```

```

END ;

```

```

RUN ;

```

```

PROC SORT DATA = FREQDAT ;

```

```

BY DUMM ;

```

```

RUN ;

```

```

*****

```

```

*** STEP 21 ***

```

```

*****

```

```

%RUNCONC (&NUM, &DIM, &DIMSQ) ;

```

```

*****

```

```

*** STEP 23 ***

```

```

*****

```

```

%TRUNKAP (&NUM, &DIM, &DIMSQ) ;

```

```

%END ;

```

```

OPTIONS FORMLIN= ;

```

```

%END RUNPROC ;

```

```

*****

```

```

*** STEP2: MACRO TO CALCULATE STUART-MAXWELL OR MCNEMAR STATISTICS ***

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*****

```

```

*** STEP 5: FILL IN ARRAY WITH COUNTS FROM PROC FREQ. ***

```

```

*** ZERO OUT MISSING VALUES

```

```

*****

```

```

DO I = 1 TO &DIM ;

```

```

DO J = 1 TO &DIM ;

```

```

IF BASE&NUM = I AND PARAM&NUM = J THEN A(I,J) = COUNT ;

```

```

IF A(I,J) = . THEN A(I,J) = 0 ;

```

```

END ;

```

```

END ;

```

```

RETAIN A1-ADDIMSQ ;

```

```

IF LAST.DUMM ;

```

```

KEEP A1-ADDIMSQ ;

```

```

RUN ;

```

```

DATA INVERT1 ;

```

```

SET INVERT ;

```

```

ARRAY A(&DIM,&DIM) A1-ADDIMSQ ;

```

```

ARRAY R(&DIM) R1-R&DIM ;

```

```

ARRAY C(&DIM) C1-C&DIM ;

```

```

CNT = 0 ;

```

```

*****

```

```

*** STEP 6: INITIALIZE TOTAL VARIABLES FOR ROWS AND COLUMNS ***

```

```

*****

```

```

DO I = 1 TO &DIM ;

```

```

R(I) = 0 ;

```

```

C(I) = 0 ;

```

```

END ;

```

```

*****

```

```

*** STEP 7: CALCULATE ROW AND COLUMN TOTALS ***

```

```

*****

```

```

DO I = 1 TO &DIM ;

```

```

DO J = 1 TO &DIM ;

```

```

R(I) = R(I) + A(I,J) ;

```

```

C(I) = C(I) + A(J,I) ;

```

```

END ;

```

```

END ;

```

```

*****

```

```

*** STEP 8: MATRIX REDUCTION SECTION TO PREVENT SINGULARITY ***

```

```

*****

```

```

*****

```

```

*** STEP 9: IF CORRESPONDING ROWS AND COLUMNS ARE ZERO, ***

```

```

*** SET THEIR ELEMENTS TO MISSING.

```

```

*****

```

```

DO I = 1 TO &DIM ;

```

```

IF C(I) = 0 AND R(I) = 0 THEN DO ;

```

```

DO J = 1 TO &DIM ;

```

```

A(I,J) = . ;

```

```

A(J,I) = . ;

```

```

END ;

```

```

CNT+1 ;

```

```

END ;

```

```

END ;

```

```

*****

```

```

*** STEP 10: RESTRUCTURE THE ARRAY ELEMENT LIST SO THAT THE ***

```

```

*** ELEMENTS OF THE ZERO ROWS & COLUMNS ARE REMOVED FROM THEIR ***

```

```

*** PRESENT POSITION AND THE REMAINING ELEMENTS SLIDE DOWN TO TAKE ***

```

```

*** THEIR PLACE. THIS ALLOWS ONE TO ADJUST THE ARRAY DIMENSION ***

```

```

*** SIZE AND PICK THE CORRECT ELEMENTS.

```

```

*****

```

```

IF CNT NE 0 THEN DO ;

```

```

I2 = 1 ;

```

```

J2 = 1 ;

```

```

DO I = 1 TO &DIM ;
  DO J = 1 TO &DIM ;
    IF A(I,J) NE . THEN DO ;
      A(I2,J2) = A(I,J) ;
      J2+1 ;
      IF MOD(J2,&DIM) = 1 THEN DO ;
        J2 = 1 ;
        I2+1 ;
      END ;
    END ;
  END ;
END ;

*****
*** STEP 11: CREATE MACRO VARIABLE WHICH ***
*** HOLDS ADJUSTED ARRAY DIMENSION ***
*****
CALL SYMPUT('NEW',&DIM - CNT) ;

RUN ;

*****
*** STEP 12: PICK THE APPROPRIATE PROC PRINT ***
*** (STUART-MAXWELL WHEN NEW > 2 OR MCNEMAR WHEN NEW = 2) ***
*****
%IF &NEW = 2 %THEN %LET PRINT = 2 ;
%IF &NEW > 2 %THEN %LET PRINT = 1 ;

*****
*** STEP 13: CREATE MACRO VARIABLES NEEDED FOR MATRIX OPERATIONS ***
*** WHICH FOLLOW, ACCORDING TO NEW ARRAY DIMENSION ***
*****
%LET DIM2 = %VAL(&NEW) ;
%LET DIMSQ2 = %VAL(&NEW**2) ;
%LET DIMPL2 = %VAR(&NEW**2) ;

*****
*** STEP 14: MCNEMAR STATISTIC CALCULATIONS FOR 2x2 TABLES ***
*** (WHEN NEW = 2), STUART-MAXWELL CALCULATIONS FOR KxK TABLES ***
*** WHERE K > 2 (WHEN NEW > 2) ***
*****
%IF &NEW = 2 %THEN %DO ;

DATA INVERT2 ;
SET INVERT1 ;

DENOM = A2 + A3 ;

IF DENOM = 0 THEN CALL SYMPUT ('PRINT', .) ;

*****
*** STEP 15: USE NORMAL APPROXIMATION CALCULATION ***
*** IF # OF DISCORDANT PAIRS (DENOM) > 20 ***
*****
ELSE IF DENOM >= 21 THEN DO ;
  MCNEMAR = (ABS(A2 - A3) - 1)**2 / DENOM ;
  PROBM = 1 - PROBCHI(MCNEMAR, 1) ;
END ;

*****
*** STEP 16: USE EXACT BINOMIAL CALCULATION IF ***
*** # OF DISCORDANT PAIRS (DENOM) < 21 ***
*****
ELSE IF A2 < (DENOM / 2) THEN DO ;
  MCNEMAR = -1 ;
  PROBM = 2 * PROBBINL(.5, DENOM, A2) ;
END ;

ELSE DO ;

```

```

  MCNEMAR = -1 ;
  PROBM = 2 * (1 - PROBBINL(.5, DENOM, A2-1)) ;
END ;

IF PROBM LE 0.05 AND PROBM NE . THEN SIGN='*' ;
FORMAT MCNEMAR MACFMT. ;

RUN ;
%END ;

%ELSE %DO ;

DATA INVERT2 ;
SET INVERT1 ;

ARRAY NN(&DIMPL2) NN1-NN&DIMPL2 ;
ARRAY D(&DIM2) D1-D&DIM2 ;
ARRAY V(&DIM2,&DIM2) V1-V&DIMSQ2 ;
ARRAY A(&DIM2,&DIM2) A1-A&DIMSQ2 ;

DO I = 1 TO &DIM2 ;
  II = &DIM2 + I ;
  NN(I) = 0 ;
  NN(II) = 0 ;
  DO J = 1 TO &DIM2 ;
    NN(I) = NN(I) + A(I,J) ;
    NN(II) = NN(II) + A(J,I) ;
  END ;
END ;

DO I = 1 TO &DIM2 ;
  II = &DIM2 + I ;
  D(I) = NN(II) - NN(I) ;
END ;

DO I = 1 TO &DIM2 ;
  II = &DIM2 + I ;
  DO J = 1 TO &DIM2 ;
    IF I EQ J THEN V(I,J) = NN(I) + NN(II) - 2*A(I,J) ;
    IF I NE J THEN V(I,J) = -(A(I,J) + A(J,I)) ;
  END ;
END ;

*****
*** STEP 17: INVERT THE COVARIANCE ***
*** MATRIX WITH THE SWEET OPERATOR ***
*****
DO K = 1 TO &DIM2-1 ;

IF V(K,K) = 0 THEN DO = . ;
ELSE DO = 1 / V(K,K) ;

V(K,K) = 1 ;

DO I = 1 TO &DIM2 ;
  V(K,I) = DO*V(K,I) ;
END ;

DO J = 1 TO &DIM2 ;
  IF J NE K THEN V(J,K) = -V(J,K)*DO ;
END ;

DO J = 1 TO &DIM2 ;
  IF J NE K THEN DO ;
    DO I = 1 TO &DIM2 ;
      IF I NE K THEN DO ;
        V(J,I) = V(J,I) + V(J,K)*V(K,I)/DO ;
      END ;
    END ;
  END ;
END ;

```

```

END ;
END ;

*****
*** STEP 18: CALCULATE CHI-SQUARE STATISTIC AND P-VALUE ***
*****
DO I = 1 TO &DIM2-1 ;
  DO J = 1 TO &DIM2-1 ;
    Q = SUM(Q, V(I,J)*D(I)*D(J)) ;
  END ;
END ;

PROBQ = 1 - PROBCHI(Q,&DIM2-1) ;
IF PROBQ LE 0.05 AND PROBQ NE . THEN SIGQ='*' ;

RUN ;
%END ;

%MACRO STU_MAX ;

*****
*** STEP 19: MACRO WHICH PICKS APPROPRIATE PROC PRINT TO USE. ***
*****
%MACRO RUNPRINT ;

OPTIONS FORMDLM=' ' ;

%IF &PRINT = 1 %THEN %DO ;

PROC PRINT DATA = INVERT2 NOOBS SPLIT = '/';
VAR Q PROBQ SIGQ ;
FORMAT Q PROBQ 8.3 ;
LABEL Q = 'STUART-MAXWELL/STATISTIC'
PROBQ = 'STUART-MAXWELL/P-VALUE'
SIGQ = '/' ;
TITLE ;
RUN ;
%END ;

%IF &PRINT = 2 %THEN %DO ;

PROC PRINT DATA = INVERT2 NOOBS SPLIT = '/';
VAR MCNEMAR PROBM SIGN ;
FORMAT MCNEMAR MACFMT. PROBM 8.3 ;
LABEL MCNEMAR = 'MCNEMAR'S TEST/EXACT BINOMIAL/OR CHI-SQ.'
PROBM = 'MCNEMAR'S TEST/ /P-VALUE'
SIGN = '/' ;
TITLE ;
RUN ;
%END ;

%END RUNPRINT ;

*****
*** STEP 21: MACRO WHICH CALCULATES AND PRINTS CONCORDANCE RATE ***
*****
%MACRO RUNCONC (NUM, DIM, DIMSQ) ;

DATA CONCORD (KEEP = TOTAL AGREE PERCENT PROB_L PROB_U) ;
SET FREQDAT ;
BY DUMM ;
IF FIRST.DUMM THEN DO ;

```

```

TOTAL = 0 ;
AGREE = 0 ;
END ;

TOTAL + COUNT ;

IF BASE&NUN = PARAM&NUN THEN AGREE + COUNT ;

RETAIN TOTAL AGREE ;

IF LAST.DUMH ;

IF TOTAL = 0 THEN DO ;
  PERCENT = . ;
  PROB_L = . ;
  PROB_U = . ;
END ;

ELSE DO ;
  PERCENT = 100*AGREE/TOTAL ;
  PROB = 1 - 0.05/2 ;

  *****
  *** STEP 22: UPPER AND LOWER 95% CONFIDENCE LIMITS FOR CONCORDANCE ***
  *** RATE ARE THE LARGE SAMPLE APPROXIMATIONS DERIVED FROM EQUIVA- ***
  *** LENT EQUATIONS GIVEN BELOW (PROB_L & PROB_U) USING THE F-DIST. ***
  *****
  N_DFL = 2*(TOTAL - AGREE + 1) ;
  D_DFL = 2*AGREE ;

  IF D_DFL = 0 THEN PROB_L = 0 ;
  ELSE DO ;
    PROBFL = FINV( PROB , N_DFL, D_DFL ) ;
    PROB_L = 100* AGREE /
      (AGREE + ( TOTAL - AGREE + 1)*PROBFL) ;
  END ;

  N_DFU = 2*(AGREE + 1) ;
  D_DFU = 2*(TOTAL - AGREE) ;

  IF D_DFU = 0 THEN PROB_U = 100 ;
  ELSE DO ;
    PROBFU = FINV( PROB , N_DFU, D_DFU ) ;
    PROB_U = 100*(1 - ((TOTAL - AGREE)/((TOTAL - AGREE)
      + (AGREE + 1)*PROBFU))) ;
  END ;

END ;

RUN ;

PROC PRINT DATA = CONCORD NOOBS LABEL ;
  VAR PERCENT PROB_L PROB_U ;
  FORMAT PERCENT PROB_L PROB_U 8.3 ;
  LABEL PERCENT = 'CONCORDANCE RATE'
  PROB_L = '95% BINOMIAL LOWER C.I.'
  PROB_U = '95% BINOMIAL UPPER C.I.' ;
TITLE ' ' ;
RUN ;

%MEND RUNCONC ;

*****
*** STEP 23: MACRO WHICH CALCULATES AND PRINTS KAPPA STATISTIC ***
*****

```

```

%MACRO RUNKAP (NUM, DIM, DIMSQ) ;

DATA KAPPA (KEEP = KAPPA UPPER LOWER Z PROBZ) ;
  SET FREEDAT ;
  BY DUMH ;

  ARRAY PCT(&DIMSQ) PCT1-PCT&DIMSQ ;
  ARRAY PCTCOL(&DIM) PCTCOL1-PCTCOL&DIM ;
  ARRAY PCTROW(&DIM) PCTROW1-PCTROW&DIM ;

  IF FIRST.DUMH THEN DO ;
    DO I = 1 TO &DIMSQ ;
      PCT(I) = 0 ;
    END ;
  END ;

  CNT = 0 ;

  DO I = 1 TO &DIM ;
    DO J = 1 TO &DIM ;
      CNT = CNT + 1 ;
      IF BASE&NUN = I AND PARAM&NUN = J THEN
        PCT(CNT) = PERCENT / 100 ;
      END ;
    END ;
  END ;

  N = SUM(N, COUNT) ;

  RETAIN PCT1-PCT&DIMSQ N ;

  IF LAST.DUMH ;

  DO I = 1 TO &DIM ;
    PCTCOL(I) = 0 ;
    PCTROW(I) = 0 ;
  END ;

  *****
  *** STEP 24: CALCULATING COLUMN MARGINAL PROBABILITIES ***
  *****
  DO I = 1 TO &DIM ;
    DO J = 0 TO &DIM - 1 ;
      PCTCOL(I) = PCTCOL(I) + PCT(I + (J * &DIM)) ;
    END ;
  END ;

  *****
  *** STEP 25: CALCULATING ROW MARGINAL PROBABILITIES ***
  *****
  CNT2 = 0 ;

  DO I = 1 TO &DIMSQ BY &DIM ;
    CNT2 = CNT2 + 1 ;
    DO J = 0 TO &DIM-1 ;
      PCTROW(CNT2) = PCTROW(CNT2) + PCT(I + J) ;
    END ;
  END ;

  *****
  *** STEP 26: CALCULATING NUMERATOR OF KAPPA STATISTIC ***
  *****
  P0 = 0 ;

  DO I = 1 TO &DIMSQ BY &DIM + 1 ;
    P0 = P0 + PCT(I) ;
  END ;

```

```

*****
*** STEP 27: CALCULATING DENOMINATOR OF KAPPA STATISTIC ***
*****
  PE = 0 ;

  DO I = 1 TO &DIM ;
    PE = PE + PCTCOL(I) * PCTROW(I) ;
  END ;

  IF PE = 1 THEN DO ;
    KAPPA = . ;
    SE = . ;
  END ;

  ELSE DO ;
    KAPPA = (P0 - PE) / (1 - PE) ;

    SUM_TERM = 0 ;

    DO I = 1 TO &DIM ;
      SUM_TERM = SUM_TERM + (PCTCOL(I) * PCTROW(I) *
        (PCTCOL(I) + PCTROW(I))) ;
    END ;
    SE = (1/((1 - PE)*SQRT(N))) * SQRT(PE + PE**2 - SUM_TERM) ;

    IF KAPPA < .000001 AND KAPPA NE . THEN KAPPA = 0 ;
    IF SE < .000001 THEN SE = . ;

    UPPER = KAPPA + 1.96 * SE ;
    LOWER = KAPPA - 1.96 * SE ;

    IF UPPER > 1 THEN UPPER = 1.000 ;

    *****
    *** STEP 28: TESTING Ho: KAPPA = 0 ***
    *****
    Z = ABS(KAPPA / SE) ;

    PROBZ = 2 * (1 - PROBNORM(Z)) ;

    RUN ;

    PROC PRINT DATA = KAPPA NOOBS LABEL ;
      VAR KAPPA LOWER UPPER Z PROBZ ;
      FORMAT KAPPA UPPER LOWER Z PROBZ 8.3 ;
      LABEL KAPPA = 'KAPPA'
      UPPER = '95% UPPER C.I.'
      LOWER = '95% LOWER C.I.'
      PROBZ = 'P-VALUE' ;
    TITLE ' ' ;
    RUN ;

  %MEND RUNKAP ;

  *****
  *** STEP 29 ***
  *****

  *****
  *** SAMPLE MACRO CALLS ***
  *****
  %RUNPROC ( A, 1, VARIABLE 1, VAR1FMT., 5) ;
  %RUNPROC ( B, 3, VARIABLE 2, VAR2FMT., 4) ;
  %RUNPROC ( C, 9, VARIABLE 4, VAR4FMT., 4) ;

```

QUIT ;

Table II.A
Clinical Assessment
Tabulation of Baseline Visit by Subsequent Visit
FOR VARIABLE 1

	SUBSEQUENT VISIT									
	HEALED		STAGE I		STAGE II		STAGE III		STAGE IV	
	N	%	N	%	N	%	N	%	N	%
BASILINE VISIT										
HEALED	0	0	0	0	0	0	0	0	0	0
STAGE I	0	0	0	0	0	0	0	0	0	0
STAGE II	71	68.93	4	3.88	26	25.24	2	1.94	0	0
STAGE III	62	51.24	0	0	15	12.40	44	36.36	0	0
STAGE IV	0	0	0	0	0	0	0	0	0	0
TOTAL	133	59.38	4	1.79	41	18.30	46	20.54	0	0

STUART-MAXWELL
STATISTIC

140.373

STUART-MAXWELL
P-VALUE

0.000

*

CONCORDANCE
RATE

31.250

95%
BINOMIAL
LOWER C.I.

25.242

95%
BINOMIAL
UPPER C.I.

37.763

KAPPA

0.146

95%
LOWER
C.I.

0.095

95%
UPPER
C.I.

0.196

Z

5.661

P-VALUE

0.000

Table II.B
Clinical Assessment
Tabulation of Baseline Visit by Subsequent Visit
FOR VARIABLE 2

	SUBSEQUENT VISIT							
	NONE		MILD		MODERATE		EXTREME	
	N	%	N	%	N	%	N	%
BASILINE VISIT								
NONE	59	89.39	7	10.61	0	0	0	0
MILD	65	86.67	10	13.33	0	0	0	0
MODERATE	0	0	0	0	0	0	0	0
EXTREME	0	0	0	0	0	0	0	0
TOTAL	124	87.94	17	12.06	0	0	0	0

MCNEMAR'S TEST
EXACT BINOMIAL
OR CHI-SQ. MCNEMAR'S TEST
P-VALUE

45.125 0.000 *

CONCORDANCE
RATE 95%
BINOMIAL
LOWER C.I. 95%
BINOMIAL
UPPER C.I.

48.936 40.431 57.487

KAPPA 95%
LOWER
C.I. 95%
UPPER
C.I. Z P-VALUE

0.026 -0.076 0.128 0.496 0.620

Table II.C
Clinical Assessment
Tabulation of Baseline Visit by Subsequent Visit
FOR VARIABLE 4

	SUBSEQUENT VISIT							
	NONE		MILD		MODERATE		SEVERE	
	N	%	N	%	N	%	N	%
BASILINE VISIT								
NONE	63	100.00	0	0	0	0	0	0
MILD	0	0	0	0	0	0	0	0
MODERATE	0	0	0	0	0	0	0	0
SEVERE	3	60.00	0	0	0	0	2	40.00
TOTAL	66	97.06	0	0	0	0	2	2.94

MCNEMAR'S TEST
EXACT BINOMIAL
OR CHI-SQ.
EXACT TEST

MCNEMAR'S TEST
P-VALUE
0.250

CONCORDANCE
RATE
95.588

95%
BINOMIAL
LOWER C.I.
87.644

95%
BINOMIAL
UPPER C.I.
99.081

KAPPA
0.553

95%
LOWER
C.I.
0.340

95%
UPPER
C.I.
0.765

Z
5.095

P-VALUE
0.000