Algorithms L.D. Fosdick Editor

Algorithm 434

Exact Probabilities for R × C Contingency Tables [G2]

David L. March [Recd. 24 Nov. 1970 and 7 Mar. 1971]
School of Education, Lehigh University,
Bethlehem, PA. 18015

Key Words and Phrases: probability, contingency table, test of significance
CR Categories: 3.5, 5.5
Language: Fortran

Description

Freeman and Halton [1] derive a general method for computing exact probabilities for contingency tables that result if a sample is subjected to k different and independent classifications. The following algorithm is limited to the case where $k = 2$.

If a sample of size N is subjected to two different and independent classifications, A and B, with R and C classes respectively, the probability P_x of obtaining the observed array of cell frequencies $X(x_{ij})$, under the conditions imposed by the arrays of marginal totals $A(r_i)$ and $B(c_j)$ is given by

$$P_x = \frac{\prod_{i=1}^{R} (r_i!) \prod_{j=1}^{C} (c_j!)}{N! \prod_{i=1}^{R} \prod_{j=1}^{C} (x_{ij}!)} \tag{1}$$

Expression (1) is exact and holds if (a) the parent population is infinite or the sampling is done with replacement of the sampled items, (b) the sampling is random, (c) the population is homogeneous, and (d) the marginal totals are considered fixed in repeated sampling.

To test the null hypothesis that A and B are independent against the indefinite two-sided alternative, the probability P_s of obtaining an array as probable as, or less probable than, the observed array is needed. P_s is found as follows: (a) the probability P_t of the observed array is computed; (b) the probabilities for all other possible arrays of cell frequencies, subject to the conditions imposed by the fixed marginal totals, are computed; and (c) P_s is then obtained by summing all of the probability values found in (b) that are less than, or equal to, the probability P_t.

Method. The method of the subroutine uses the fact that expression (1) can be rewritten as

$$P_x = Q_x/R_x$$

where

$$Q_x = \frac{\prod_{i=1}^{R} (r_i!) \prod_{j=1}^{C} (c_j!)}{N!}$$

which is constant for the given set of marginal totals (r_i) and (c_j) and

$$R_x = \prod_{i=1}^{R} \prod_{j=1}^{C} (x_{ij}!)$$

which varies depending on the array of cell frequencies (x_{ij}). In order to avoid machine overflow and roundoff error, these computations are performed using logarithms.

The observed $R \times C$ contingency table is specified by the $NR \times NC$ matrix which is partitioned as follows:

$$\begin{array}{cccc|c} x_{11} & \cdots & \cdots & x_{1C} & r_1 \\ \vdots & & & & \vdots \\ x_{R1} & \cdots & \cdots & x_{RC} & r_R \\ \hline c_1 & \cdots & \cdots & c_C & N \end{array}$$

After computing the constant term $QXLOG$ and the probability of the given table PT, the subroutine assigns to each of the lower right $(R - 1) \times (C - 1)$ cells the minimum of its corresponding row and column totals which is the maximum possible number for the cell. These cells are then varied in all possible combinations with each cell varied between its maximum number and zero.

Starting with cell (2,2), the variation is accomplished by subtraction of 1. When the subtraction yields a zero or positive result the routine goes to compute the remainder of the cell frequencies. When a negative result is obtained, the cell in question, say cell (i, j), is reset to the minimum of the corresponding row and column totals, 1 is subtracted from cell $(i, j + 1)$ or, if $j + 1$ is greater than C, cell $(i + 1, 2)$, and the count down resumes at cell (2,2). If none of the lower right $(R - 1) \times (C - 1)$ cells yield a zero or positive result, the computations are complete and the subroutine returns to the caller. For example, if the top line (below) is the cell maximum ordered left to right from the (2,2) to the (R, C) cell, the combinations generated will be

$$\begin{array}{cccc} 2 & 1 & 1 & \cdots \\ 1 & 1 & 1 & \cdots \\ 0 & 1 & 1 & \cdots \\ 2 & 0 & 1 & \cdots \\ 1 & 0 & 1 & \cdots \\ 0 & 0 & 1 & \cdots \\ 2 & 1 & 0 & \cdots \\ & \vdots & & \\ 0 & 0 & 0 & \cdots \end{array}$$

The column 1 and row 1 cells are filled by subtraction of the generated cell numbers from the marginal totals. Since the method described above yields illegal as well as legal partitions, it is possible to obtain a negative result for one of these cells. When this occurs, the routine goes back to get a new set of cell frequencies. Otherwise

RXLOG is computed. Then, the probability *PX* is computed and added to the cumulative sum *PC*. If *PX* is less than, equal to, or, to avoid missing one due to computational inaccuracy, slightly larger than *PT*, *PX* is also added to the significance probability *PS*.

Since *PC* is the probability of obtaining some of the tables possible within the constraints of the marginal totals, *PC* should equal 1.0. . The accuracy of the result can be estimated from the amount of deviation of *PC* from 1.0. .

The floating point logarithms (base 10) of the integer factorials are obtained from function *FACLOG*. For arguments less than or equal to 100, the result is obtained from a table that is computationally filled on the first reference to *FACLOG*. Stirling's approximation is used for arguments greater than 100.

Results. The algorithm was tested on a CDC 6400 (60 bit word) using 2×3 ($N = 30$), 2×4 ($N = 7$), and 3×3 ($N = 7$) contingency tables. Results for the 2×3 tables were verified against values separately computed using programs developed by March [2]. In several cases *PC* deviated from 1.0. by 1.0×10^{-12}. Results for the 2×4 and 3×3 tests were verified by hand computation.

The author is indebted to the referees for their valuable comments and suggestions.

References

1. Freeman, G.H., and Halton, J.H. Note on an exact treatment of contingency, goodness of fit, and other problems of significance. *Biometrika 38* (1951), 141–149.
2. March, D.L. Accuracy of the chi-square approximation for 2×3 contingency tables with small expectations. An unpublished D.Ed. Diss., School of Education, Lehigh U., Bethlehem, Pa., 1970.

Algorithm

```
      SUBRØUTINE CØNP(MATRIX,NR,NC,PT,PS,PC)
C
C INPUT ARGUMENTS.
C
C     MATRIX = SPECIFICATIØN ØF THE CØNTINGENCY TABLE.
C         THIS MATRIX IS PARTITIØNED AS FØLLØWS
C
C              X(11).....X(1C)   R(1)
C                .  .....  .       .
C                .  .....  .       .
C              X(R1).....X(RC)   R(R)
C               C(1)..... C(C)    N
C
C         WHERE X(IJ) ARE THE ØBSERVED CELL FREQUENCIES,
C         R(I) ARE THE RØW TØTALS, C(J) ARE THE CØLUMN
C         TØTALS, AND N IS THE TØTAL SAMPLE SIZE.
C         NØTE THAT THE ØRIGINAL CELL FREQUENCIES ARE
C         DESTRØYED BY THIS SUBRØUTINE.
C
C     NR = THE NUMBER ØF RØWS IN MATRIX (R=NR-1).
C
C     NC = THE NUMBER ØF CØLUMNS IN MATRIX (C=NC-1).
C
C ØUTPUT ARGUMENTS.
C
C     PT = THE PRØBABILITY ØF ØBTAINING THE GIVEN TABLE.
C
C     PS = THE PRØBABILITY ØF ØBTAINING A TABLE AS PRØBABLE
C         AS, ØR LESS PRØBABLE THAN, THE GIVEN TABLE.
C
C     PC = THE PRØBABILITY ØF ØBTAINING SØME ØF THE
C         TABLES PØSSIBLE WITHIN THE CØNSTRAINTS ØF THE
C         MARGINAL TØTALS. (THIS SHØULD BE 1.0.  DEVIATIØNS
C         FRØM 1.0 REFLECT THE ACCURACY ØF THE CØMPUTATIØN.)
C
C EXTERNALS.
C
C     FACLØG(N) = FUNCTIØN TØ RETURN THE FLØATING PØINT
C         VALUE ØF LØG BASE 10 ØF N FACTØRIAL.
C
      DIMENSIØN MATRIX(NR,NC)
      INTEGER R,C,TEMP
C
      R=NR-1
      C=NC-1
C
C CØMPUTE LØG ØF CØNSTANT NUMERATØR
C
      QXLØG=-FACLØG(MATRIX(NR,NC))
      DØ 10 I=1,R
   10   QXLØG=QXLØG+FACLØG(MATRIX(I,NC))
      DØ 20 J=1,C
   20   QXLØG=QXLØG+FACLØG(MATRIX(NR,J))
C
C CØMPUTE PRØBABILITY ØF GIVEN TABLE
C
      RXLØG=0.0
      DØ 50 I=1,R
      DØ 50 J=1,C
   50   RXLØG=RXLØG+FACLØG(MATRIX(I,J))
      PT=10.0**(QXLØG-RXLØG)
C
      PS=0.0
      PC=0.0
C
C FILL LØWER RIGHT (R-1) X (C-1) CELLS WITH
C MINIMUM ØF RØW AND CØLUMN TØTALS
C
      DØ 100 I=2,R
      DØ 100 J=2,C
  100   MATRIX(I,J)=MINO(MATRIX(I,NC),MATRIX(NR,J))
      GØ TØ 300
C
C ØBTAIN A NEW SET ØF FREQUENCIES IN
C LØWER RIGHT (R-1) X (C-1) CELLS
C
  200 DØ 220 I=2,R
      DØ 220 J=2,C
        MATRIX(I,J)=MATRIX(I,J)-1
        IF(MATRIX(I,J).GE.0) GØ TØ 300
  220   MATRIX(I,J)=MINO(MATRIX(I,NC),MATRIX(NR,J))
      RETURN
C
C FILL REMAINDER ØF ØBSERVED CELLS
C .....CØMPLETE CØLUMN 1
C
  300 DØ 320 I=2,R
        TEMP=MATRIX(I,NC)
        DØ 310 J=2,C
  310     TEMP=TEMP-MATRIX(I,J)
        IF(TEMP.LT.0) GØ TØ 200
  320   MATRIX(I,1)=TEMP
C
C .....CØMPLETE RØW 1
C
      DØ 340 J=1,C
        TEMP=MATRIX(NR,J)
        DØ 330 I=2,R
  330     TEMP=TEMP-MATRIX(I,J)
        IF(TEMP.LT.0) GØ TØ 200
  340   MATRIX(1,J)=TEMP
C
C CØMPUTE LØG ØF THE DENØMINATØR
C
      RXLØG=0.0
      DØ 350 I=1,R
      DØ 350 J=1,C
  350   RXLØG=RXLØG+FACLØG(MATRIX(I,J))
C
C CØMPUTE PX.  ADD TØ PS IF PX .LE. PT
C (ALLØW FØR RØUND-ØFF ERRØR)
C
      PX=10.0**(QXLØG-RXLØG)
      PC=PC+PX
      IF((PT/PX).GT.0.99999) PS=PS+PX
      GØ TØ 200
      END
      FUNCTIØN FACLØG(N)
C
C INPUT ARGUMENT.
C
C      N = AN INTEGER GREATER THAN ØR EQUAL TØ ZERØ.
C
C FUNCTIØN RESULT.
C
C      FACLØG = THE LØG TØ THE BASE 10 ØF N FACTØRIAL.
C
      DIMENSIØN TABLE(101)
      DATA TPILØG/0.39908 99342/
      DATA ELØG  /0.43429 44819/
      DATA IFLAG/0/
C
C USE STIRLINGS APPRØXIMATIØN IF N GT 100
C
      IF(N.GT.100) GØ TØ 50
C
C LØØK UP ANSWER IF TABLE WAS GENERATED
C
      IF(IFLAG.EQ.0) GØ TØ 100
   10 FACLØG=TABLE(N+1)
      RETURN
C
C HERE FØR STIRLINGS APPRØXIMATIØN
C
   50 X=FLØAT(N)
      FACLØG=(X+0.5)*ALØG10(X) - X*ELØG + TPILØG
     1     + ELØG/(12.0*X) - ELØG/(360.0*X*X*X)
      RETURN
C
C HERE TØ GENERATE LØG FACTØRIAL TABLE
C
  100 TABLE(1)=0.0
      DØ 120 I=2,101
        X=FLØAT(I-1)
  120   TABLE(I)=TABLE(I-1)+ALØG10(X)
      IFLAG=1
      GØ TØ 10
      END
```