Appl. Statist. (1991)
40, *No.* 1, *pp.* 195–236

Algorithm AS 260

Evaluation of the Distribution of the Square of the Sample Multiple-correlation Coefficient

By Cherng G. Ding†

National Chiao Tung University, Taipei, Republic of China

and Rolf E. Bargmann

University of Georgia, Athens, USA

[Received July 1988. Revised October 1989]

Keywords: Multiple-correlation coefficient; Non-central beta distribution

Language

Fortran 77

Description and Purpose

Let $X_1, \ldots, X_p$ be distributed as $N_p(\mu, \Sigma)$ and R be the sample multiple-correlation coefficient between X_i and the other $p-1$ random variables based on a sample of size N. The density of R^2 is given by (see, for example, Anderson (1984))

$$\frac{(1-R^2)^{(N-p-2)/2}(R^2)^{(p-1)/2-1}(1-\rho^2)^{(N-1)/2}}{\Gamma\{(N-1)/2\}\ \Gamma\{(N-p)/2\}} \sum_{k=0}^{\infty} \frac{(\rho^2)^k(R^2)^k\ \Gamma^2\{(N-1)/2+k\}}{k!\ \Gamma\{(p-1)/2+k\}}, \quad (1)$$

where ρ denotes the population multiple-correlation coefficient. The function subprogram SQMCOR computes the cumulative distribution function (CDF) of R^2.

Numerical Method

Let $X=R^2$, $a=(p-1)/2$ and $b=(N-p)/2$. Then the CDF of X is given by

$$M(x; a, b, \rho^2) = \sum_{k=0}^{\infty} q(k; a, b, \rho^2)\ I_x(a+k, b), \quad (2)$$

where $I_x(a, b)$ is the central beta(a, b) CDF and

$$q(k; a, b, \rho^2) = \Gamma(a+b+k)\ (\rho^2)^k(1-\rho^2)^{a+b}/\{k!\ \Gamma(a+b)\}.$$

Note that $0 \leqslant x \leqslant 1$, $0 \leqslant \rho^2 \leqslant 1$, $p > 1$, $N > p$ and both p and N are integers. Lenth (1987) has given an algorithm for computing the non-central beta CDF which is a modification of one given in Norton (1983). The advantage is that only one

†*Address for correspondence*: Institute of Management Science, National Chiao Tung University, 4F, 114 Chung-Hsiao W. Road, Section 1, Taipei, Taiwan, Republic of China.

0035–9254/91/40195 $2.00

evaluation of the incomplete beta function is required instead of several, and hence computation time is greatly saved. The non-central beta CDF is a series of central beta distributions with Poisson weights. Since the CDF of X can be expressed in the same way but with different weights (negative binomial if N is odd), the method used to develop the algorithm will basically follow that given by Lenth. Let E_n be the error of truncation in equation (2) after the first $n+1$ terms. Using the well-known relation for the incomplete beta function (see, for example, Abramowitz and Stegun (1965), p. 944),

$$I_x(a+1, b) = I_x(a, b) - \frac{\Gamma(a+b)}{\Gamma(a+1)\,\Gamma(b)}\, x^a(1-x)^b, \tag{3}$$

and the fact that $\Sigma_{k=0}^{\infty} q(k; a, b, \rho^2) = 1$ (see, for example, Abramowitz and Stegun (1965), p. 556), we have

$$\begin{aligned} E_n &= \sum_{k=n+1}^{\infty} q(k; a, b, \rho^2)\, I_x(a+k, b) \\ &< I_x(a+n+1, b) \sum_{k=n+1}^{\infty} q(k; a, b, \rho^2) \\ &= I_x(a+n+1, b)\left\{1 - \sum_{k=0}^{n} q(k; a, b, \rho^2)\right\}. \end{aligned} \tag{4}$$

Recursive iterations are performed until the error bound above is less than a predetermined small number. The central beta CDF (incomplete beta functions) in equation (2) can be obtained by first computing $I_x(a, b)$ and then using equation (3).

Structure

REAL FUNCTION SQMCOR(X, IP, N, RHO2, IFAULT)

Formal parameters

X	Real	input:	the percentage point R^2 (at which the cumulative probability is desired)
IP	Integer	input:	the number of random variables p
N	Integer	input:	the sample size N
RHO2	Real	input:	the square of the population multiple-correlation coefficient ρ^2
IFAULT	Integer	output:	a fault indicator: $=1$ if $n>$ ITRMAX and $E_n>$ ERRMAX (see the section on constants); $=2$ if $p \leqslant 1$, $p \geqslant N$, $\rho^2<0$ or $\rho^2>1$; $=3$ if $x<0$ or $x>1$; $=0$ otherwise

Auxiliary Algorithms

Function ALOGAM (Pike and Hill, 1966) is used to compute the natural logarithm of the gamma function; alternatively ALNGAM (Macleod, 1989) could be used.

Function BETAIN (Majumder and Bhattacharjee, 1973) is used to compute the incomplete beta function.

Constants

The variables ERRMAX and ITRMAX are set by the DATA statement in SQMCOR. ERRMAX denotes a bound on the error of truncation. The value given here is 1.0×10^{-6}. ITRMAX is an integer to control the number of iterations. The value given here is 100.

Precision

Double-precision operation may be performed by changing REAL to DOUBLE PRECISION in the FUNCTION statement and in the REAL statement. The real constants in the DATA statements also need to be converted to double precision, and the value of ERRMAX may be changed from 1.0×10^{-6} to 1.0×10^{-12}. Moreover, similar modifications to the auxiliary routines ALOGAM and BETAIN are required.

Time

The execution time is a non-decreasing function of x and ρ^2.

Restrictions

For some unusual situations, the series may not converge within ITRMAX (e.g. 100) iterations. Then the value of ITRMAX in the DATA statement can be enlarged to obtain the results with the precision required.

References

Abramowitz, M. and Stegun, I. A. (1965) *Handbook of Mathematical Functions*. New York: Dover Publications.

Anderson, T. W. (1984) *An Introduction to Multivariate Statistical Analysis*, 2nd edn, p. 145. New York: Wiley.

Lenth, R. V. (1987) Algorithm AS 226: Computing noncentral beta probabilities. *Appl. Statist.*, **36**, 241–244.

Macleod, A. J. (1989) Algorithm AS 245: A robust and reliable algorithm for the logarithm of the gamma function. *Appl. Statist.*, **38**, 397–402.

Majumder, K. L. and Bhattacharjee, G. P. (1973) Algorithm AS 63: The incomplete beta integral. *Appl. Statist.*, **22**, 409–411.

Norton, V. (1983) A simple algorithm for computing the non-central *F* distribution. *Appl. Statist.*, **32**, 84–85.

Pike, M. C. and Hill, I. D. (1966) Algorithm 291: Logarithm of gamma function. *Communs Ass. Comput. Mach.*, **9**, 684.


```
      REAL FUNCTION SQMCOR(X, IP, N, RHO2, IFAULT)
C
C        ALGORITHM AS 260  APPL. STATIST. (1991) VOL. 40, NO. 1
C
C        Computes the C.D.F. for the distribution of the
C        square of the multiple correlation coefficient
C        with parameters IP, N, and RHO2
```

```
C
C        The following auxiliary algorithms are required:
C        ALOGAM - Log-gamma function (CACM 291)
C         (or  ALNGAM    AS 245)
C        BETAIN - Incomplete beta function (AS 63)
C
      REAL X, RHO2
      INTEGER IP, N, IFAULT
      REAL A, B, BETA, ERRBD, ERRMAX, GX, Q, SUMQ, TEMP, TERM, XJ,
     *     ZERO, HALF, ONE
      INTEGER ITRMAX
      REAL ALOGAM, BETAIN
      EXTERNAL ALOGAM, BETAIN
C
      DATA ERRMAX, ITRMAX / 1.0E-6, 100 /
      DATA ZERO, HALF, ONE / 0.0, 0.5, 1.0 /
C
      SQMCOR = X
      IFAULT = 2
      IF (RHO2 .LT. ZERO .OR. RHO2 .GT. ONE .OR. IP .LT. 2 .OR.
     *    N .LE. IP) RETURN
      IFAULT = 3
      IF (X .LT. ZERO .OR. X .GT. ONE) RETURN
      IFAULT = 0
      IF (X .EQ. ZERO .OR. X .EQ. ONE) RETURN
C
      A = HALF * (IP - 1)
      B = HALF * (N - IP)
C
C        Initialize the series
C
      BETA = EXP(ALOGAM(A, IFAULT) + ALOGAM(B, IFAULT) -
     *       ALOGAM(A + B, IFAULT))
      TEMP = BETAIN(X, A, B, BETA, IFAULT)
C
C        There is no need to test IFAULT since all of the
C        parameter values have already been checked
C
      GX = EXP(A * LOG(X) + B * LOG(ONE - X) - LOG(A)) / BETA
      Q = (ONE - RHO2) ** (A + B)
      XJ = ZERO
      TERM = Q * TEMP
      SUMQ = ONE - Q
      SQMCOR = TERM
C
C        Perform recurrence until convergence is achieved
C
   10 XJ = XJ + ONE
      TEMP = TEMP - GX
      GX = GX * (A + B + XJ - ONE) * X / (A + XJ)
      Q = Q * (A + B + XJ - ONE) * RHO2 / XJ
      SUMQ = SUMQ - Q
      TERM = TEMP * Q
      SQMCOR = SQMCOR + TERM
C
C        Check for convergence and act accordingly
C
      ERRBD = (TEMP - GX) * SUMQ
      IF ((INT(XJ) .LT. ITRMAX) .AND. (ERRBD .GT. ERRMAX)) GO TO 10
      IF (ERRBD .GT. ERRMAX) IFAULT = 1
      RETURN
      END
```