Algorithm AS 190: Probabilities and Upper Quantiles for the Studentized Range

Author(s): R. E. Lund and J. R. Lund

Source: *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 1983, Vol. 32, No. 2 (1983), pp. 204-210

Published by: Wiley for the Royal Statistical Society

Stable URL: https://www.jstor.org/stable/2347300

```
      IF (I .LE. LM(1)) RNL = RNL + FLOAT(RL(I, 1)) * ALOG(MEW + A)
      IF (I .LE. LM(2)) RNL = RNL + FLOAT(RL(I, 2)) * ALOG(1.0 - MEW +
     *  A)
      RNL = RNL - FLOAT(RL(I, 3)) * ALOG(1.0 + A)
    5 CONTINUE
      RETURN
      END
      SUBROUTINE GDER(MEW, THETA, RL, MRL, LM, IDER, RD, PD)
C
C        ALGORITHM AS 189.4  APPL. STATIST. (1983) VOL.32, NO.2
C
C        GENERAL DERIVATIVE SUBROUTINE
C
      REAL MEW, THETA, PD(IDER), A, B, C, D
      INTEGER RL(MRL, 3), LM(3), RD(2, IDER)
      MLM = LM(3)
      KK = IDER - 1
      DO 5 I = 1, IDER
    5 PD(I) = 0.0
      DO 45 I = 1, MLM
      C = FLOAT(I - 1)
      A = C * THETA
      DO 40 J = 1, 3
      IF (I .GT. LM(J)) GOTO 40
      GOTO (10, 15, 20), J
   10 D = MEW + A
      GOTO 25
   15 D = 1.0 - MEW + A
      GOTO 25
   20 D = 1.0 + A
   25 B = FLOAT(RL(I, J)) / D ** KK
      IF (J .EQ. 3) GOTO 35
      DO 30 K = 1, IDER
      PD(K) = PD(K) + FLOAT(RD(J, K)) * B
      B = B * C
   30 CONTINUE
      GOTO 40
   35 D = -FLOAT(RD(1, 1)) * B * C ** KK
      PD(IDER) = PD(IDER) + D
   40 CONTINUE
   45 CONTINUE
      RETURN
      END
```

Algorithm AS 190

Probabilities and Upper Quantiles for the Studentized Range

By R. E. LUND† and J. R. LUND

Montana State University, Bozeman, Montana, USA

[Received January 1981. Revised January 1983]

Keywords: MULTIPLE COMPARISONS; SIMULTANEOUS CONFIDENCE INTERVALS; STUDENTIZED RANGE

LANGUAGE

Fortran 66

DESCRIPTION AND PURPOSE

The ratio $Q = \text{range}/(\text{standard deviation}) = W/S$ is commonly called a studentized range. The

† *Present address*: Dept of Maths – MSU, College of Letters and Science, Bozeman, MT. 59717, USA.

 0035–9254/83/32204 $2.00

random variable Q is distributed as the studentized range probability distribution whenever W is the range for R independent normal variates having common mean μ and variance σ^2 and the ratio VS^2/σ^2 is an independent chi-squared variate having V degrees of freedom.

A typical application is to produce R sample means by sampling R populations independently with samples of size N. Under an assumption of homogeneous variances, S is calculated as the square root of the within-sample mean square divided by N. The degrees of freedom, V is given by $R(N-1)$. Under the null hypothesis of equal population means, Q is distributed as the studentized range probability distribution. Tables for the distribution are used to formulate multiple comparisons and to construct simultaneous confidence intervals (Miller, 1966).

Most texts on statistical methods contain a table of quantiles for the studentized range, usually only at probability level 0.95. Harter (1969) produced an extensive set of tables for which he claimed six-decimal accuracy in probability.

The purpose of this algorithm is to supplement such tables by approximating probabilities for 0 to Q for arbitrary values of Q, V and R and, given P in the interval (0.90, 0.99) and arbitrary V and R, to approximate the quantile Q. The constants controlling accuracy within the program are set to achieve approximately three decimal place accuracy on the probability scale. The algorithm is expected to have value for integration into computer programs producing simultaneous confidence intervals, multiple comparisons and calculating achieved significance levels for sample ranges.

Numerical Method for Probability Calculations

The probability integral from zero to Q can be written as

$$P(Q, V, R) = C \int_0^\infty x^{V-1} e^{-Vx^2/2} \{R \int_{-\infty}^\infty \theta(y)\, [\Phi(y) - \Phi(y - Qx)]^{R-1} dy\}\, dx \qquad (1)$$

where

$$C = \frac{V^{V/2}}{\Gamma\left(\frac{V}{2}\right) 2^{V/2-1}},$$

$$\theta(y) = (1/\sqrt{(2\pi)})\, e^{-y^2/2},$$

and

$$\Phi(y) = \int_{-\infty}^{y} \theta(t)\, dt.$$

When the degrees of freedom V are infinite the probability integral takes the simpler form

$$P(Q, R) = R \int_{-\infty}^\infty \theta(y)\, [\Phi(y) - \Phi(y - Q)]^{R-1} dy \qquad (2)$$

The inner integral in (1) (the K Do-loop in *PRTRNG*) is computed using a composite trapezoid rule and the outer integral (the J Do-loop) is computed using a sine function quadrature rule (Stenger, 1973). Before the computation begins the node associated with the contribution to the maximum probability is located approximately by maximizing the discretized approximation of P as a function of GK and HJ where GK and HJ are the nodes in the trapezoid and sine quadrature rules, respectively. The summation of probability progresses in all directions from the approximate maximum until the values contributed drop below the cut-off values *PCUTK* and *PCUTJ*. A minimum number of steps is specified in each direction to prevent cut-off prior to achieving the maximum when the starting location is poorly located with respect to the maximum. The step sizes, G and H, are varied as functions of V and R to obtain uniform accuracy. (*GK*, *HJ*, G and H are real variables, in *PRTRNG*.)

Numerical Method for Quantiles

An initial approximate quantile Q_1 is produced by *FUNCTION QTRNGO* which utilizes a polynomial function of V, R and T where T is the quantile of Student's t distribution associated with the user specified P. Quantile T is produced by an approximation method due to Peiser (1943) which utilizes a normal quantile produced by *FUNCTION GAUINV* (Odeh and Evans, 1974). A call to *FUNCTION PRTRNG* evaluates the probability P_1 corresponding to Q_1. Knowledge of P_1, in turn, enables one to use *QTRNGO* to produce a refined approximation Q_2. The secant method, which requires two initial values, is then applied iteratively to the error function $F(Q) = P - P_i$ to reach the desired level of accuracy if this accuracy has not already been attained by Q_1 and Q_2.

STRUCTURE

FUNCTION PRTRNG(Q, V, R, IFAULT)
FUNCTION QTRNG(P, V, R, IFAULT)

Formal parameters

P	Real	input:	lower side cumulative probability, $0.90 \leqslant P \leqslant 0.99$
Q	Real	input:	upper limit of studentized range, lower limit is zero
V	Real	input:	degrees of freedom, $V \geqslant 1.0$
R	Real	input:	number of samples, $R \geqslant 2.0$
IFAULT	Integer	output:	failure indicator,

Failure indications

IFAULT = 0 is no error
= 1 if $V < 1.0$ or $R < 2.0$
= 2 if $P < 0.90$ or $P > 0.99$ in *QTRNG*
= 9 if an error occurs in *FUNCTION GAUINV* used by *QTRNG*. The error in *GAUINV* cannot occur with current settings of constants

Auxiliary algorithms

FUNCTION ALNORM(X, UPPER) – Algorithm AS 66 (Hill, 1973).
FUNCTION GAUINV(P, IFAULT) – Algorithm AS 70 (Odeh and Evans, 1974).

Other algorithms may be substituted for *ALNORM* but the user must be aware that accuracy in *PRTRNG* is strongly dependent upon accuracy of such normal probabilities. However, accuracy in the extreme tails is not essential. An additional statement may be inserted in *ALNORM* to omit actual calculation whenever $|X| > 3.62$ but assign 0.0 to *ALNORM* for $X < -3.62$ and 1.0 for $X > 3.62$. This change produces a negligible loss in accuracy to three decimals but provides a significant time reduction. The particular substitution for the statement labelled 10 in *ALNORM* is

```
10 IF (Z.GT.–3.62 AND. Z.LT.3.62) GOTO 20
```

ACCURACY

The accuracy of *PRTRNG* is evaluated by a comparison with the tables published by Harter (1969) for the settings of all combinations of $P = 0.001$, 0.01, 0.05, 0.10, 0.50, 0.90, 0.95, 0.99, 0.999, $V = 2$, 8, 30, 120, ∞ and $R = 2$, 8, 30, 100. Harter claimed accuracy to six decimals. A maximum difference (or error) of 0.0054 was obtained for the rarely encountered combination of $P = 0.10$, $V = 2$ and $R = 100$, for the constants used in the present program. All but four of the

180 settings tested produced differences less than 0.0005. Nearly four decimal place accuracy was achieved for all cases with $V \geqslant 8$.

The accuracy of *QTRNG* was established by following its call by one additional call to the *FUNCTION PRTRNG* to validate the probability of the quantile produced. A test run was based on all combinations of the parameter levels of $P = 0.90, 0.95, 0.99$; $V = 2, 8, 30, 120, \infty$ and $R = 2, 8, 30, 100$. Differences or errors in probability were less than 0.0005 in nearly all cases. It should be noted that *PRTRNG* also contains a maximum error of about 0.0005 for these values of the parameters. Accuracy testing utilized 36 bit real variables.

Constants influencing accuracy

The accuracy and execution time of the quadrature rule used in *PRTRNG* is strongly influenced by the fineness of the node spacing. The constant *STEP* controls the node spacing in both the x (J in the program with step H) and y (K in the program with step G) directions. The minimum (maximum) number of nodes is controlled in the x and y directions by *JMIN* (*JMAX*) and *KMIN* (*KMAX*), respectively. The actual truncation of the numerical quadrature rule is controlled within the limits defined by *JMIN*, *JMAX*, *KMIN* and *KMAX* by measuring the probability contributed at each node. *PCUTJ* is the lower limit for the inclusion of pairs of nodes symmetric about the midpoint in the x direction and *PCUTK* is the lower limit of pairs of rows of nodes (the y direction). That is: given a row K, the procedure starts at the midpoint of the row and works outward in calculating the probability at the nodes symmetric with respect to the midpoint. Rows are then computed in a symmetric outward fashion (increasing $|y|$).

Any alterations in *PRTRNG* of the limits *JMIN*, *JMAX*, *KMIN* and *KMAX*, the arrays $QW(\cdot)$ and $VW(\cdot)$ and $(STEP)^{-1}$ should be performed proportionally. Any increase in the value of *JMAX* requires the dimension of each of the arrays *QW* and *VW* to be increased to at least twice the new value of *JMAX*. However, to obtain large improvements in accuracy both a conversion to double precision in arithmetic operations as well as a reduction in *PCUTJ* and *PCUTK* would be required.

If the use of *PRTRNG* is restricted to cases for which $P \geqslant 0.90$ and $V \geqslant 8$, approximately three digit accuracy can be achieved by resetting the constants to $JMIN = 2$, $KMIN = 4$ and $STEP = 0.9$. Such a constant setting reduces execution time by about two-thirds. If this setting is used for all P, but with $V \geqslant 8$, errors become as high as 0.01. Allowing V to be as low as 2 can produce errors as high as 0.05. The errors are of size approximately proportional to P for $P < 0.5$ and $1 - P$ for $P > 0.5$.

In *QTRNG*, the constants *JMAX* and *PCUT* were selected to produce about three decimal place accuracy in the probability corresponding to the approximated quantile. *JMAX* (which has a minimum value of two) controls the maximum number of calls to *PRTRNG*. *PCUT* controls termination of the refinement process prior to reaching *JMAX*. Any adjustments in accuracy in *PRTRNG* will lead to corresponding alterations of accuracy in *QTRNG*.

TIMING

Execution time for *PRTRNG* can be approximated by determination of the elementary operations performed. The median number of nodes evaluated was 135 for the current constant setting for the test combinations treated in the section on accuracy. The evaluation at each node requires approximately six additions and a single each of multiplication, exponentiation, logarithm computation and a call to *ALNORM*. The call to *ALNORM* uses at least seven additions and five multiplication-divisions.

Operation time for *QTRNG* perhaps can best be specified as a multiple of the number of calls to *FUNCTION PRTRNG* which is the major time-using procedure. The test run referenced in the section on accuracy required a median of only two calls to *PRTRNG*. About one-sixth of the approximations reached the required accuracy level with only one call of *PRTRNG* while another one-sixth required three or four calls to *PRTRNG*.

ADDITIONAL COMMENTS

The function *QTRNGO* may have independent use for fast but inexact quantiles for P in the interval of 0.80 to 0.99. For the test run referenced in the section on accuracy, only one-sixth of the initial quantiles exceeded an error of 0.02 on the probability scale and no errors exceeded 0.05. Errors are approximately proportional to $1-P$. The largest errors occur for $V=2$. If the case of $V=2$ was eliminated when attempting to obtain quantiles for 0.99, the largest error was 0.006 on the probability scale.

RELATED ALGORITHM

Dunlap *et al.* (1977) published an algorithm for probabilities for the studentized range. It requires an evaluation of the integrand at 1032 nodes to produce accuracy to about four decimals. He excluded the troublesome case of $V<3$ which is allowed herein.

REFERENCES

Dunlap, W. P., Powell, R. S. and Konnerth, T. K. (1977) A FORTRAN IV function for calculating probabilities associated with the studentized range statistic. *Behaviour Research Methods and Instrumentation*, 9, 373–375.

Harter, H. L. (1969) *Order Statistics and Their Use in Testing and Estimation*, Vol. 1, Washington: U.S. Government Printing Office.

Hill, I.D. (1973) Algorithm AS 66. The normal integral. *Appl. Statist.*, **22**, 424–427.

Miller, R. J. Jr. (1966) *Simultaneous Statistical Inference*. New York: McGraw-Hill.

Odeh, R. E. and Evans, J. O. (1974) Algorithm AS 70. The percentage points of the normal distribution. *Appl. Statist.*, **23**, 96–97.

Peiser, A. M. (1943) Asymptotic formulas for significance levels of certain distributions. *Ann. Math. Statist.*, **14**, 56–62. [and correction *Ann. Math. Statist.*, **20**, 128–129 (1949).]

Stenger, F. (1973) Integration formula based on the trapezoid formula. *J. Inst. Maths. Applics.*, **12**, 103–114.

```
      FUNCTION PRTRNG(Q, V, R, IFAULT)
C
C        ALGORITHM AS 190  APPL. STATIST. (1983) VOL.32, NO.2
C
C        EVALUATES THE PROBABILITY FROM 0 TO Q FOR A STUDENTIZED
C        RANGE HAVING V DEGREES OF FREEDOM AND R SAMPLES.
C
C        USES FUNCTION ALNORM - ALGORITHM AS 66.
C
C        ARRAYS VW AND QW STORE TRANSIENT VALUES USED IN THE
C        QUADRATURE SUMMATION.   NODE SPACING IS CONTROLLED BY
C        STEP.   PCUTJ AND PCUTK CONTROL TRUNCATION.
C        MINIMUM AND MAXIMUM NUMBER OF STEPS ARE CONTROLLED BY
C        JMIN, JMAX, KMIN AND KMAX.   ACCURACY CAN BE INCREASED
C        BY USE OF A FINER GRID - INCREASE SIZES OF ARRAYS VW
C        AND QW, AND JMIN, JMAX, KMIN, KMAX AND 1/STEP PROPORTIONALLY.
C
      REAL Q, V, R, VW(30), QW(30), PCUTJ, PCUTK, STEP, VMAX, ZERO,
     *  FIFTH, HALF, ONE, TWO, CV1, CV2, CVMAX, CV(4)
      DATA PCUTJ, PCUTK, STEP, VMAX /0.00003E0, 0.0001E0, 0.45E0,
     *  120.0E0/, ZERO, FIFTH, HALF, ONE, TWO /0.0E0, 0.2E0, 0.5E0,
     *  1.0E0, 2.0E0/, CV1, CV2, CVMAX /0.193064705E0, 0.293525326E0,
     *  0.39894228E0/, CV(1), CV(2), CV(3), CV(4) /0.318309886E0,
     *  -0.268132716E-2, 0.347222222E-2, 0.833333333E-1/
      DATA JMIN, JMAX, KMIN, KMAX /3, 15, 7, 15/
C
C        CHECK INITIAL VALUES.
C
      PRTRNG = ZERO
      IFAULT = 0
      IF (V .LT. ONE .OR. R .LT. TWO) IFAULT = 1
      IF (Q .LE. ZERO .OR. IFAULT .EQ. 1) GOTO 99
C
C        COMPUTING CONSTANTS, LOCATING MIDPOINT, ADJUSTING STEPS.
```

```
      G = STEP * R ** (-FIFTH)
      GMID = HALF * ALOG(R)
      R1 = R - ONE
      C = ALOG(R * G * CVMAX)
      IF (C .GT. VMAX) GOTO 20
C
      H = STEP * V ** (-HALF)
      V2 = V * HALF
      IF (V .EQ. ONE) C = CV1
      IF (V .EQ. TWO) C = CV2
      IF (.NOT.(V .EQ. ONE .OR. V .EQ. TWO)) C = SQRT(V2) * CV(1)
     *  / (ONE + ((CV(2) / V2 + CV(3)) / V2 + CV(4)) / V2)
      C = ALOG(C * R * G * H)
C
C         COMPUTING INTEGRAL.
C         GIVEN A ROW K, THE PROCEDURE STARTS AT THE MIDPOINT AND WORKS
C         OUTWARD (INDEX J) IN CALCULATING THE PROBABILITY AT NODES
C         SYMMETRIC ABOUT THE MIDPOINT.   THE ROWS (INDEX K) ARE ALSO
C         PROCESSED OUTWARDS SYMMETRICALLY ABOUT THE MIDPOINT.   THE
C         CENTRE ROW IS UNPAIRED.
C
   20 GSTEP = G
      QW(1) = -ONE
      QW(JMAX + 1) = -ONE
      PK1 = ONE
      PK2 = ONE
      DO 28 K = 1, KMAX
      GSTEP = GSTEP - G
   21 GSTEP = -GSTEP
      GK = GMID + GSTEP
      PK = ZERO
      IF (PK2 .LE. PCUTK .AND. K .GT. KMIN) GOTO 26
      W0 = C - GK * GK * HALF
      PZ = ALNORM(GK, .TRUE.)
      X = ALNORM(GK - Q, .TRUE.) - PZ
      IF (X .GT. ZERO) PK = EXP(W0 + R1 * ALOG(X))
      IF (V .GT. VMAX) GOTO 26
C
      JUMP = -JMAX
   22 JUMP = JUMP + JMAX
      DO 24 J = 1, JMAX
      JJ = J + JUMP
      IF (QW(JJ) .GT. ZERO) GOTO 23
      HJ = H * FLOAT(J)
      IF (J .LT. JMAX) QW(JJ + 1) = -ONE
      EHJ = EXP(HJ)
      QW(JJ) = Q * EHJ
      VW(JJ) = V * (HJ + HALF - EHJ * EHJ * HALF)
C
   23 PJ = ZERO
      X = ALNORM(GK - QW(JJ), .TRUE.) - PZ
      IF (X .GT. ZERO) PJ = EXP(W0 + VW(JJ) + R1 * ALOG(X))
      PK = PK + PJ
      IF (PJ .GT. PCUTJ) GOTO 24
      IF (JJ .GT. JMIN .OR. K .GT. KMIN) GOTO 25
   24 CONTINUE
   25 H = -H
      IF (H .LT. ZERO) GOTO 22
C
   26 PRTRNG = PRTRNG + PK
      IF (K .GT. KMIN .AND. PK .LE. PCUTK .AND. PK1 .LE. PCUTK) GOTO 99
      PK2 = PK1
      PK1 = PK
      IF (GSTEP .GT. ZERO) GOTO 21
   28 CONTINUE
C
   99 RETURN
      END
C
      FUNCTION QTRNG(P, V, R, IFAULT)
C
C         ALGORITHM AS 190.1  APPL. STATIST. (1983) VOL.32, NO.2
```

```
C        APPROXIMATES THE QUANTILE P FOR A STUDENTIZED RANGE
C        DISTRIBUTION HAVING V DEGREES OF FREEDOM AND R SAMPLES
C        FOR PROBABILITY P, P.GE.0.90 .AND. P.LE.0.99.
C
C        USES FUNCTIONS  ALNORM,GAUINV,PRTRNG AND QTRNG0 -
C        ALGORITHMS AS 66, AS 70, AS 190 AND AS 190.2
C
      REAL P, V, R, PCUT, P75, P80, P90, P99, P995, P175, ONE, TWO, FIVE
      DATA JMAX, PCUT, P75, P80, P90, P99, P995, P175, ONE, TWO, FIVE /
     *  8, 0.001E0, 0.75E0, 0.80E0, 0.90E0, 0.99E0, 0.995E0, 1.75E0,
     *  1.0E0, 2.0E0, 5.0E0/
C
C        CHECK INPUT PARAMETERS
C
      IFAULT = 0
      NFAULT = 0
      IF (V .LT. ONE .OR. R .GT. TWO) IFAULT = 1
      IF (P .LT. P90 .OR. P .GT. P99) IFAULT = 2
      IF (IFAULT .NE. 0) GOTO 99
C
C        OBTAIN INITIAL VALUES
C
      Q1 = QTRNG0(P, V, R, NFAULT)
      IF (NFAULT .NE. 0) GOTO 99
      P1 = PRTRNG(Q1, V, R, NFAULT)
      IF (NFAULT .NE. 0) GOTO 99
      QTRNG = Q1
      IF (ABS(P1 - P) .LT. PCUT) GOTO 99
      IF (P1 .GT. P) P1 = P175 * P - P75 * P1
      IF (P1 .LT. P) P2 = P + (P - P1) * (ONE - P) / (ONE - P1) * P75
      IF (P2 .LT. P80) P2 = P80
      IF (P2 .GT. P995) P2 = P995
      Q2 = QTRNG0(P2, V, R, NFAULT)
      IF (NFAULT .NE. 0) GOTO 99
C
C        REFINE APPROXIMATION
C
      DO 14 J = 2, JMAX
      P2 = PRTRNG(Q2, V, R, NFAULT)
      IF (NFAULT .NE. 0) GOTO 99
      E1 = P1 - P
      E2 = P2 - P
      QTRNG = (E2 * Q1 - E1 * Q2) / (E2 - E1)
      IF (ABS(E1) .LT. ABS(E2)) GOTO 12
      Q1 = Q2
      P1 = P2
   12 IF (ABS(P1 - P) .LT. PCUT * FIVE) GOTO 99
      Q2 = QTRNG
   14 CONTINUE
C
   99 IF (NFAULT .NE. 0) IFAULT = 9
      RETURN
      END
C
      FUNCTION QTRNG0(P, V, R, IFAULT)
C
C        ALGORITHM AS 190.2  APPL. STATIST. (1983) VOL.32, NO.2
C
C        CALCULATES AN INITIAL QUANTILE P FOR A STUDENTIZED RANGE
C        DISTRIBUTION HAVING V DEGREES OF FREEDOM AND R SAMPLES
C        FOR PROBABILITY P, P.GT.0.80.AND.P.LT.0.995.
C
C        USES FUNCTION GAUINV - ALGORITHM AS 70
C
      REAL P, V, R, Q, T, VMAX, HALF, ONE, FOUR, C1, C2, C3, C4, C5
      DATA VMAX, HALF, ONE, FOUR, C1, C2, C3, C4, C5 /120.0E0, 0.5E0,
     *  1.0E0, 4.0E0, 0.8843E0, 0.2368E0, 1.214E0, 1.208E0, 1.4142E0/
C
      T = GAUINV(HALF + HALF * P, IFAULT)
      IF (V .LT. VMAX) T = T + (T * T * T + T) / V / FOUR
      Q = C1 - C2 * T
      IF (V .LT. VMAX) Q = Q - C3 / V + C4 * T / V
      QTRNG0 = T * (Q * ALOG(R - ONE) + C5)
      RETURN
      END
```