TINT: A Microsoft BASIC t integration program

D. LOUIS WOOD

University of Arkansas, Little Rock, Arkansas

and

DIANNE WOOD

Little Rock School District, Little Rock, Arkansas

Researchers seeking exact parametric probabilities for publication or other precision purposes generally encounter two thorny problems. First, areas under *t*, chi-square, *F*, and other curves must be determined with high accuracy. The second problem is that these parametric probability density functions include gamma functions that must be resolved exactly before numerical integration can begin. These take the form of $\Gamma(n/2)$ in chi-square and *t* and also $\Gamma[(n+1)/2]$ in *t*. Exact resolution of these functions is straightforward when *n* < 58; however, with more degrees of freedom, most 8-bit microcomputers, including the TRS-80 Model 4 we use, overflow.

The present program, TINT (see Listing 1), effectively addresses both problems and illustrates the process with a two-tailed significance test for the *t* distribution.

Numerical integration of the density function of *t* is accomplished with high accuracy via a fast-running algorithm of Simpson's rule. Like other curve-area-finding techniques, Simpson's rule (Anton, 1981) divides the area to be assessed into small segments. Unlike many other techniques (e.g., trapezoidal rule), the tops of the segments curve and thus fit the *t* or other curve closely. This results in highly accurate area determinations per number of segments employed. The number of segments required for any specified level of accuracy is a function of the range of numerical integration. For values of *t* = 5 or less, the program employs 60 segments; for $5 < t \leq 15$, 240 segments are employed. In those rarely seen cases in which *t* > 15, the program sets the number of segments according to the *t* and the degrees of freedom.

The *t* density function shown here (CRC Standard Mathematical Tables, Beyer, 1979, p. 523) includes $\Gamma[(n+1)/2]/\Gamma(n/2)$, and that ratio provides the key for the gamma resolution subroutines employed in TINT:

$$f(t) = \frac{\Gamma[(n+1)/2]}{\sqrt{n\pi} \Gamma(n/2) [1 + (t^2/n)]^{.5(n+1)}}$$

The first author's mailing address is: Department of Psychology, University of Arkansas, 33rd Street and University Avenue, Little Rock, AR 72204. The second author's mailing address is: Supervisor of Mathematics, Little Rock School District, Markham and Izard Streets, Little Rock, AR 72201.

Specifically, $\Gamma[(n+1)/2]/\Gamma(n/2)$ is simplified by recognizing that, since *n* is integral, both gamma functions are factorial processes containing common factors that can be divided out. The result in TINT is *G1/G*, which is divided by $\sqrt{n\pi}$ to form the parameter *C*. Because the common terms are divided out before multiplication, exact gamma resolution obtains up to *df* = 113 without overflow. At *df* = 114, the TRS-80 Model 4 and Model III both overflow, but at *df* = 115, accurate results are again achieved, and that is the upper limit of the routine. When *C* is computed, it is incorporated into the Simpson's rule segment and probabilities are determined.

Writing formulas for the remaining factors that *G1* and *G* comprise is complicated by the fact that these algorithms differ according to the following conditions:

- (1) *n* is even and evenly divisible by 4 (*n*+1 is odd).
- (2) *n* is even and not evenly divisible by 4 (*n*+1 is odd).
- (3) *n* is odd (*n*+1 is even and evenly divisible by 4).
- (4) *n* is odd (*n*+1 is even and not evenly divisible by 4).

In each of the four cases, *G1* and *G* are products, as indicated in Table 1.

When *n* < 5, the values of *G1* and *G* have been written into the program, with all values except *G1* for *df* = 4 being taken from the CRC Tables (Beyer, 1979, pp. 413-414); *G1* for *df* = 4 was computed with the programmed routine.

The program begins by requesting the number of degrees of freedom for a problem and, upon entry, resolves the gamma functions and computes *C*. It next requests *t* and inputs it as *ABS(t)*. The limits of numerical integration are then set at zero and *ABS(t)*. That area is doubled and is printed as "Area Within"; it is the probability of chance occurrence of a *t* having a smaller abso-

Table 1
Gamma Function Resolution in *t* Integration

Case	G1	G
1	$[(a_1/4)(a_2/4) \dots (a_n/4)]\sqrt{\pi}$, where <i>a_j</i> are consecutive odd integers, <i>a₁</i> = (<i>n</i> /2) + 1, and <i>a_n</i> = <i>n</i> - 1	(2) (4) ... [(<i>n</i> /2) - 2]
2	$[(a_1/4)(a_2/4) \dots (a_n/4)]2\sqrt{\pi}$, where <i>a_j</i> are consecutive odd integers, <i>a₁</i> = <i>n</i> /2, and <i>a_n</i> = <i>n</i> - 1	(2) (4) ... [(<i>n</i> -1)/2]
3	(2) (4) ... [(<i>n</i> -1)/2]	$[(a_1/4)(a_2/4) \dots (a_n/4)]\sqrt{\pi}$, where <i>a_j</i> are consecutive odd integers, <i>a₁</i> = (<i>n</i> +1)/2, and <i>a_n</i> = <i>n</i> - 2
4	(2) (4) ... [(<i>n</i> -3)/2]	$[(a_1/4)(a_2/4) \dots (a_n/4)]2\sqrt{\pi}$, where <i>a_j</i> are consecutive odd integers, <i>a₁</i> = (<i>n</i> +3)/2, <i>a_n</i> = <i>n</i> - 2

lute value than the one entered. The two-tailed probability is given by $(1 - \text{Area Within})$ and is printed as "Area Beyond".

The accuracy of the routine depends upon the level of precision into which variables and parameters are defined, the degrees of freedom, and the number of segments into which the range is subdivided. Testing reveals that the routine is almost always exact to six decimals, but on rare occasions the sixth digit is incorrect, prompting the conclusion that exact results are provided until the computed integral (Area Within) exceeds .999990. At this point, the two-tailed probability is .000010, which defines the limit of accuracy of the program. Beyond its range of accuracy, the routine can even give negative probabilities because Simpson's rule can overestimate, and a value of "Area Within" that is slightly larger than 1 yields a very small negative two-tailed probability.

The accuracy of computing extremely small probabilities can be enhanced by defining all variables in double precision and listing all parameters at 16 decimal places. We do not recommend this procedure, however, because it runs very slowly, probabilities in the millionths are rarely of value, and we have not exhaustively tested this configuration. The program is optimally run as published within the restrictions mentioned.

Running times vary as a function of the number of segments involved. All runs with $df > 3$ and those with $df \leq 3$ and $t < 51$ require from 3 to 15 seconds on the TRS-80 Model 4. The only very slow numerical integrations occur when $df = 1, 2$, or 3 and t is very large. These range from 30 sec when $50 < t < 106$ to a maximum of 8 min 25 sec when $df = 1$ and $636 < t < 2001$. Such values of t are rarely encountered, but can be of pedagogical or theoretical interest.

The program is written in Level II BASIC and was developed and tested on a TRS-80 Model 4 computer. It also runs verbatim on the Model III. Because no commands specific to TRS-80 computers were used, it should have wide applicability with minimal changes.

Availability: A listing of the program can be obtained

```

1 CLS:PRINT"Two-Tailed Integration of t":FOR J=1 TO 560:NEXT
2 DEFINT H,J,K,M,N,W:INPUT "ENTER df":N:K=N+1
3 GOSUB 15:C=(G1/G)*(1/SQR(N*3.14159265#))
4 INPUT"ENTER t":TT:U=ABS(TT):PRINT "
5 IF TT>5 THEN GOSUB 42 ELSE W=30
6 PRINT"C O M P U T I N G !":PRINT "
7 PRINT"Area Beyond", "Area Within":PRINT "
8 D=U/W:GOSUB 41:S=S+Y:X=X+D:GOSUB 41:S=S+Y:M=W/2
9 FOR J=1 TO M:X=(2+J-1)*D:GOSUB 41:T=T+Y:NEXT:S=S+4*T:T=0
10 FOR J=1 TO M-1:X=2+J*D:GOSUB 41:T=T+Y:NEXT:S=S+2*T
11 A1=1-2*D*S/3:A2=2*D*S/3:PRINT USING"#####";ABS(A1);PRINT,, A2
12 PRINT"For Another Problem Press The Spacebar;Otherwise <Break>
13 Z$=INKEY$:IF Z$="" THEN 13
14 CLEAR:PRINT " ":GOTO 2
15 IF N>4 THEN 21
16 ON N GOTO 17,18,19,20
17 G=1.7724538#:G1=1:RETURN
18 G=1:G1=.886227:RETURN
19 G=.886227:G1=1:RETURN
20 G=1:G1=.329340428#:RETURN
21 IF N/2=INT(N/2) THEN CC=1:GOTO 29
22 IF (N-1)/4>INT((N-1)/4) THEN 26
23 LL=2:UL=(N-1)/2:GOSUB 35:G1=G
24 LL=(N+1)/2:UL=N-2:GOSUB 39
25 RETURN
26 LL=2:UL=(N-3)/2:GOSUB 35:G1=G
27 LL=(N+3)/2:UL=N-2:GOSUB 36
28 RETURN
29 IF N/4>INT(N/4) THEN 33
30 LL=N/2+1:UL=N-1:GOSUB 39:G1=G
31 LL=2:UL=N/2-2:GOSUB 35
32 RETURN
33 LL=N/2:UL=N-1:GOSUB 36:G1=G
34 LL=2:UL=N/2-1
35 G=1:FOR J=LL TO UL STEP 2:G=G*J:NEXT:RETURN
36 G=1:FOR J=LL TO UL STEP 2:G=G*J/4:NEXT
37 IF CC=1 AND N/4>INT(N/4) THEN G=G*3.54491:RETURN
38 G=G*.886227:RETURN
39 G=1:FOR J=LL TO UL STEP 2:G=G*J/4:NEXT
40 G=G*1.77245:RETURN
41 Y=C/SQR((1+X*X/N)^K):RETURN
42 IF TT<15 THEN W=120 ELSE 44
43 PRINT"This Computation Is Somewhat Slow":PRINT " ":RETURN
44 IF N>7 THEN 52
45 PRINT"This Computation Is Very Slow;Please Be Patient.":PRINT " "
46 IF TT<51 THEN W=240:RETURN
47 IF N>3 THEN 52 ELSE IF TT<106 THEN W=480:RETURN
48 IF N>2 THEN 52 ELSE IF TT<201 THEN W=960:RETURN
49 IF N=2 AND TT<301 THEN W=1030:RETURN
50 IF N=2 THEN 52 ELSE IF TT<637 THEN W=3000:RETURN
51 IF TT<2001 THEN W=8004:RETURN
52 CLS:PRINT"THE RANGE OF ACCURACY HAS BEEN EXCEEDED.":END

```

Listing 1. TINT program for two-tailed integration of the t distribution.

without charge from D. Louis Wood, Department of Psychology, University of Arkansas, 33rd Street and University Avenue, Little Rock, Arkansas 72204.

REFERENCES

- ANTON, H. (1981). *Calculus*. New York: Wiley.
 BEYER, W. H. (Ed.). (1979). *CRC standard mathematical tables* (25th ed.). Boca Raton, FL: CRC Press.

(Revision accepted for publication October 24, 1984.)