ALGORITHM 15

EVALUATION OF PROBABILITY
FOR THE F-SNEDCOR DISTRIBUTION

1. Function declaration.

real procedure PF_Snedcor($n$, $m$, $FO$, error);
value $n$, $m$, $FO$;
integer $n$, $m$;
real $FO$;
label error;
comment $PF_Snedcor$ calculates the probability $P(F > FO)$,
where $FO$ is a given value of the F-Snedcor function.
Data:
$FO$ — value of F-Snedcor test function,
$n$ — number of degrees of freedom for the first random
variable (in the numerator of the F-formula),
$m$ — number of degrees of freedom for the second random
variable (in the denominator of the F-formula).
Other parameters:
error — label indicating error exit ($n < 1$ or $m < 1$ or $FO < 0$);

begin
integer $i$, $k$;
real $a$, $PF$, $RF$;
Boolean $B$;
switch $E$: = $pp$, $pn$, $np$, $np$;
if $n < 1 \lor m < 1 \lor FO < .0$ then go to error;
$B$: = true;
$PF$: = .0;
go to if $1.0 + FO = 1.0$ then FIN
else $E$[$2 \times (n-n \div 2 \times 2-m \div 2) + m + 1$];

$pp$: if $m \leq n$ then go to $np$;
$pn$: $k$: = $n$;
$n$: = $m$;
$m$: = $k$;
\[ FO: = 1.0/FO; \]
\[ B: = \text{false}; \]
\[ np: \quad FO: = m/(m+n \times FO); \]
\[ \text{if } m = 1 \text{ then go to mone; } \]
\[ i: = m\div2; \]
\[ k: = i+i; \]
\[ a: = RF: = PF: = (1.0-FO) \uparrow (n/k); \]
\[ n: = n-2; \]
\[ m: = m\div2; \]
\[ \text{for } k: = m-k+4 \text{ step 2 until } m \text{ do } \]
\[ \begin{align*}
& \text{begin} \\
& \quad RF: = (n+k) \times FO \times RF/k; \\
& \quad PF: = PF + RF \\
& \text{end } k; \\
& RF: = PF \times a \uparrow (i-1); \\
& \text{if } m\div2 \times 2 = m \text{ then go to FIN; } \\
& n: = n+2; \\
& PF: = n \times PF \times \sqrt{FO}; \\
& \text{mone: } FO: = 1.0-FO; \\
& RF: = .0; \\
& a: = \sqrt{(FO-FO\times FO)}; \\
& \text{for } k: = n \text{ step } -2 \text{ until } 3 \text{ do } \\
& \quad \begin{align*}
& \quad RF: = (FO-FO/k) \times RF + a; \\
& \quad PF: = PF - PF/k \\
& \text{end } k; \\
& PF: = 0.5 - (2.0 \times (RF-PF) - \arctan((FO-0.5)/a)) \times 0.31830988618379\ldots = 1.0/\pi; \\
& \text{FIN: if } PF > 1.0 \text{ then } PF: = 1.0; \\
& PF/Snedecor: = \text{if } B \text{ then } 1.0-PF \text{ else } PF \\
& \text{end } PF/Snedecor \\
\end{align*} \]

2. Method used. The well-known formula for the probability of the \( F \)-Snedecor distribution has the form
\[
P(F > F_0) = n^{n/2}m^{m/2} \int_{F_0}^\infty F^{n/2-1} (nF+m) - (n+m)/2 dF/B(n/2, m/2),
\]
where

\[ B(p, q) = \int_0^1 t^{p-1}(1-t)^{q-1} \, dt. \]

Substituting \( x = nF/(nF_0 + m), \) \( x_0 = nF_0/(nF_0 + m), \) and writing

\[ Q(n, m, x) = \int_0^x t^{n/2-1}(1-t)^{m/2-1} \, dt, \]

we have \( B(n/2, m/2) = Q(n, m, 1) \) and

\begin{equation}
(1) \quad P(F > F_0) = 1 - Q(n, m, x_0)/Q(n, m, 1).
\end{equation}

Let us write

\[ P(n, m, x) = x^{n/2} \sum_{k=0}^{m/2-1} \frac{(n+2k-2)!!}{(n-2k)!!(2k)!!} \frac{(1-x)^k}{(n-2k)!!(2k+1)!!} \]

\[ R(n, m, x) = 2\sqrt{x(1-x)} \sum_{k=0}^{m/2-1} \frac{(2k)!!}{(2k+1)!!} x^k + \arctan \frac{0.5-x}{\sqrt{x(1-x)}} \]

and

\[ H(n, m, x) = \begin{cases} 
P(n, m, x) & \text{for } m \text{ even,} \\
1 - P(m, n, 1-x) & \text{for } n \text{ even,} \\
0.5 + [R(n, m, x) - 2 \frac{(n-1)!!}{(n-2)!!} \sqrt{1-x} P(n, m, x)]/\pi & \text{for } n, m \text{ odd.}
\end{cases} \]

With the recurrent formulas

\[ Q(n, 2, x) = 2x^{n/2}/n \quad \text{for } n = 1, 2, \ldots, \]

\[ Q(1, 1, x) = \arctan \frac{0.5-x}{\sqrt{x(1-x)}}. \]

\[ Q(k, l, x) = \begin{cases} 
x^{1/2}(1-x)^{l/2} + (k/2 - 1) Q(k-2, l, x))/(k/2 + l/2 - 1) & \text{for } k = 3, 4, \ldots, \\
x^{1/2}(1-x)^{l/2-1} + (l/2 - 1) Q(k, l-2, x))/(k/2 + l/2 - 1) & \text{for } l = 3, 4, \ldots,
\end{cases} \]

we may prove that

\begin{equation}
(2) \quad Q(n, m, x) = Q(n, m, 1)^* H(n, m, x).
\end{equation}

Finally, from (1) and (2) it follows that

\[ P(F > F_0) = 1 - H(n, m, x). \]
3. **Certification.** \( PFSnedecor \) calculates the same value as \( F_{test} \) [3] or Fisher [1]. All three procedures were tested and compared on the Odra 1013 computer with 31-bit floating-point mantissa, in the FALA-69 autocode. It appears that

1° \( PFSnedecor \) is much faster than Fisher and in some cases faster than \( F_{test} \).

2° When one of the degrees of freedom is greater than 120, Fisher gives floating-point overflow. When one of the degrees of freedom is greater than 128, \( F_{test} \) uses some approximative formulae, giving a lesser accuracy. \( PFSnedecor \) uses exact formulae for all \( n \) and \( m \).

3° \( F_{test} \) is 2 times longer than \( PFSnedecor \), uses 2 times more variables, and must be complemented by a procedure Gauss [2]. Moreover, \( F_{test} \) exploits the functions \( \sin \) and \( \cos \).

**References**


**Received on 4.11.1970; revised version on 8.3.1971**

**S. GRZEGÓRSKI (Lublin)**

**OBLICZANIE PRAWDOPODOBIĘSTWA DLA ROZKŁADU F-SNEDECORA**

**STRESZCZENIE**

Procedura \( PFSnedecor \) oblicza prawdopodobieństwo \( P(F > F0) \) dla dowolnych liczb stopni swobody, wykorzystując wzory dokładne.

Dane:

- \( F0 \) — wartość funkcji testowej \( F \)-Snedecora,
- \( n \) — liczba stopni swobody zmiennej występującej w liczniku,
- \( m \) — liczba stopni swobody zmiennej występującej w mianowniku.

Przez zastosowanie tej procedury, proces weryfikowania hipotez wykorzystujących rozkład \( F \)-Snedecora sprowadza się do obliczenia prawdopodobieństwa i ewentualnego porównania go z ustalonym poziomem ufności. Nie ma zatem potrzeby korzystania z tablic rozkładu \( F \)-Snedecora.