APPENDIX 1

This Maple program computes the p-values (right hand tail) for the noncentral generalized F distribution and plots the density of this distribution

Donald Ramirez

Mathematics Department

University of Virginia

der@virginia.edu

April 1, 2004

```
> restart:
```

Initialized the statistics packages

```
> with(stats):
> with(statevalf):
> with(pdf):
> with(cdf):
```

```
Warning, these names have been redefined: beta, cauchy, chisquare,
exponential, fratio, gamma, laplaced, logistic, lognormal, normald,
studentst, uniform, weibull
```

Input initial values

s = degrees of freedom for the numerator of the generalized F distribution

nu = degrees of freedom for the denominator of the generalized F distribution

alpha = vector of length s of positive weights

omega = vector of length s of noncentrality parameters

y0 = value to compute the p-value; that is Pr[W > y0]

```
> s := 2:
> nu := 15:
> alpha :=  [1.6335, 1.9300]:
> omega :=  [1.2757, 1.3537]:
> y0 := statevalf[icdf,fratio[s,nu]](0.95);
```

$$y0 := 3.682320344$$

User defined parameters

max_N is maximum number of terms in the partial sum that will be retained

error_tolerance controls truncation error

```
> max_N := 40:
> error_tolerance := 10^(-5):
```

Start of procedure

```
> noncentrality := sum( omega['i']^2, 'i' = 1 ..  s);
```

$$noncentrality := 3.45991418$$

procedure sum_dc computes the required coefficients c[k]

```
> sum_dc := proc (k)
> local temp, i;
> global c, d;
> temp := d(k)*c[0];
> for i to k-1 do temp := temp+d(k-i)*c[i] end do;
> c[k] := simplify(temp/k)
> end proc;
```

$sum\_dc := \mathbf{proc}(k)$
$\mathbf{local}\, temp,\, i;$
$\mathbf{global}\, c,\, d;$
$temp := \mathrm{d}(k) * c_0\,;$
$\mathbf{for}\, i\, \mathbf{to}\, k-1\, \mathbf{do}\;\; temp := temp + \mathrm{d}(k-i) * c_i\, \mathbf{end\ do}\,;$
$c_k := \mathrm{simplify}(temp/k)$
$\mathbf{end\ proc}$

procedure compute_c computes the required coefficients d[k] and returns the number of terms in the partial sum to meet the error tolerance

```
>  compute_c := proc (temp_alpha)
>  local i, temp_A, A, temp_d, k, temp, temp1_alpha :
>  global b, c, d, N, s, nu, omega, noncentrality:
>  temp1_alpha := sort(temp_alpha):
>  b := .99*temp1_alpha[1]:
>  temp_A := sqrt(product(b/temp_alpha['i'],('i') = 1 ..  s)):
>  A := temp_A*exp(-1/2*noncentrality):
>  temp_d :=
>  (1/2)*sum((1-b/temp_alpha['i'])^k+k*b*omega['i']^2*(1-b/temp_alpha['i'
>  ])^(k-1)/temp_alpha['i'],('i') = 1 ..  s):
>  d := unapply(temp_d, k):
>  c[0] := A:
>  for k from 1 to max_N do sum_dc(k) end do:
>  k := 0:
>  temp := c[0];
>  while (s/(b*(s+2*(k+1)))*(1-temp) > error_tolerance and k < max_N)
do
>  k := k + 1;
>  temp := sum(c['i'], 'i' = 0 ..  k):
>  end do:
>  N := k:  end proc;
```

compute_c := **proc**(*temp_alpha*)
local *i*, *temp_A*, *A*, *temp_d*, *k*, *temp*, *temp1_alpha*;
global *b*, *c*, *d*, *N*, *s*, ν, ω, *noncentrality*;
temp1_alpha := sort(*temp_alpha*) ;
b := 0.99 ∗ *temp1_alpha*$_1$;
temp_A := sqrt(product($b/$*temp_alpha*$_{,_i}$, '*i*' = 1..*s*)) ;
A := *temp_A* ∗ exp(−*noncentrality*/2) ;
temp_d := 1/2 ∗ sum((1 − $b/$*temp_alpha*$_{,_i}$)^k
+ *k* ∗ *b* ∗ $\omega_{,_i}^2$ ∗ (1 − $b/$*temp_alpha*$_{,_i}$)$^{(k-1)}/$*temp_alpha*$_{,_i}$, '*i*' = 1..*s*);
d := unapply(*temp_d*, *k*) ;
c_0 := *A* ;
for *k* **to** *max_N* **do** sum_dc(*k*) **end do** ;
k := 0 ;
temp := c_0 ;
while *error_tolerance* < *s* ∗ (1 − *temp*)/(*b* ∗ (*s* + 2 ∗ *k* + 2)) **and** *k* < *max_N* **do**
k := *k* + 1 ; *temp* := sum($c_{,_i}$, '*i*' = 0..*k*)
end do;
N := *k*
end proc

procedure gen_F_pdf computes the required partial sum for the density of the generalized F distribution

```
> gen_F_pdf := proc(y)
> local temp, k:
> temp := 0;
> for k from 0 to N do temp := temp + (s/nu)*(c[k]/b)*GAMMA((2*k
+ nu +
> s)/2)*((s/nu)*y/b)^((s + 2*k - 2)/2)/ (GAMMA((s +
> 2*k)/2)*GAMMA(nu/2)*(1 + (s/nu)*y/b)^((2*k + nu + s)/2))
> end do;
> end proc;
```

$$gen\_F\_pdf := \mathbf{proc}(y)$$
$$\mathbf{local}\ temp,\ k;$$
$$temp := 0\,;$$
$$\mathbf{for}\ k\ \mathbf{from}\ 0\ \mathbf{to}\ N\ \mathbf{do} temp := temp + s * c_k * \Gamma(k + \nu/2 + s/2) *$$
$$(s * y/(\nu * b))^{(s/2+k-1)}/(\nu * b * \Gamma(s/2 + k) * \Gamma(\nu/2) * (1 + s * y/(\nu * b))^{(k+\nu/2+s/2)})$$
$$\mathbf{end\ do}$$
$$\mathbf{end\ proc}$$

compute number of terms to be used

```
> number_of_terms_used := compute_c(alpha);
```

$$number\_of\_terms\_used := 10$$

```
> gen_F_cdf := x -> evalf(Int(gen_F_pdf(t), t= 0 ..  x)):
```

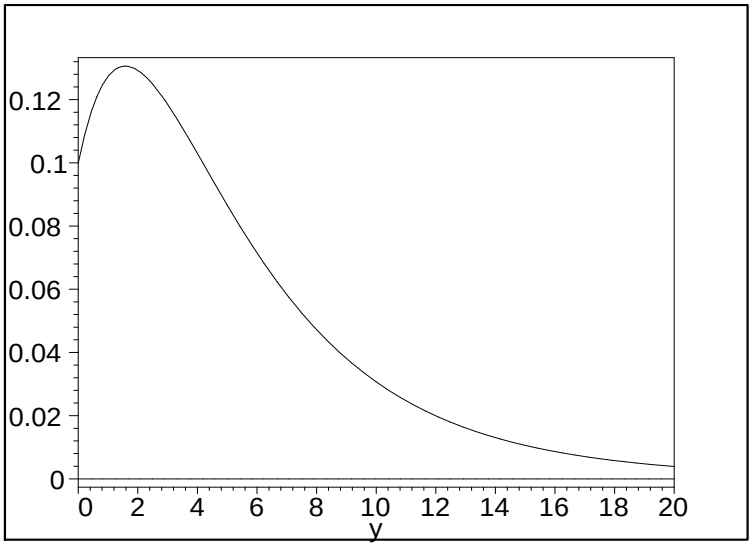
compute the p-value the value y0; that is Pr[W > y0]

```
> p_value := 1-gen_F_cdf(y0);
```

$$p\_value := 0.5517948918$$

plot the density of the non-central generalized F distribution

```
> plot({gen_F_pdf(y), 0} , y = 0 ..  20, thickness = 2, axes =
> 'boxed', color = 'black');
```


0.12
0.1
0.08
0.06
0.04
0.02
0
0 2 4 6 8 10 12 14 16 18 20
y