NAG Toolbox

nag_specfun_bessel_k1_real (s18ad)

1 Purpose

nag_specfun_bessel_k1_real (s18ad) returns the value of the modified Bessel function $K_1(x)$, via the function name.

2 Syntax

```
[result, ifail] = nag_specfun_bessel_k1_real(x)
[result, ifail] = s18ad(x)
```

3 Description

nag_specfun_bessel_k1_real (s18ad) evaluates an approximation to the modified Bessel function of the second kind $K_1(x)$.

Note: $K_1(x)$ is undefined for $x \leq 0$ and the function will fail for such arguments.

The function is based on five Chebyshev expansions:

For $0 < x \leq 1$,

$$K_1(x) = \frac{1}{x} + x \ln x \sum_{r=0} a_r T_r(t) - x \sum_{r=0} b_r T_r(t), \qquad \text{where } t = 2x^2 - 1.$$

For $1 < x \leq 2$,

$$K_1(x) = e^{-x} \sum_{r=0} c_r T_r(t), \qquad \text{where } t = 2x - 3.$$

For $2 < x \leq 4$,

$$K_1(x) = e^{-x} \sum_{r=0} d_r T_r(t), \qquad \text{where } t = x - 3.$$

For $x > 4$,

$$K_1(x) = \frac{e^{-x}}{\sqrt{x}} \sum_{r=0} e_r T_r(t), \qquad \text{where } t = \frac{9 - x}{1 + x}.$$

For x near zero, $K_1(x) \simeq \frac{1}{x}$. This approximation is used when x is sufficiently small for the result to be correct to ***machine precision***. For very small x on some machines, it is impossible to calculate $\frac{1}{x}$ without overflow and the function must fail.

For large x, where there is a danger of underflow due to the smallness of K_1, the result is set exactly to zero.

4 References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

5 Parameters

5.1 Compulsory Input Parameters

1: **x** – REAL (KIND=nag_wp)

The argument x of the function.

Constraint: $\mathbf{x} > 0.0$.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **result**

The result of the function.

2: **ifail** – INTEGER

$\mathbf{ifail} = 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

$\mathbf{ifail} = 1$

$\mathbf{x} \leq 0.0$, K_1 is undefined. On softfailure the function returns zero.

$\mathbf{ifail} = 2$ (*warning*)

x is too small, there is a danger of overflow. On softfailure the function returns approximately the largest representable value.

$\mathbf{ifail} = -99$

An unexpected error has been triggered by this routine. Please contact NAG.

$\mathbf{ifail} = -399$

Your licence key may have expired or may not have been installed correctly.

$\mathbf{ifail} = -999$

Dynamic memory allocation failed.

7 Accuracy

Let δ and ϵ be the relative errors in the argument and result respectively.

If δ is somewhat larger than the ***machine precision*** (i.e., if δ is due to data errors etc.), then ϵ and δ are approximately related by:

$$\epsilon \simeq \left| \frac{xK_0(x) - K_1(x)}{K_1(x)} \right| \delta.$$

Figure 1 shows the behaviour of the error amplification factor

$$\left| \frac{xK_0(x) - K_1(x)}{K_1(x)} \right|.$$

However if δ is of the same order as the ***machine precision***, then rounding errors could make ϵ slightly larger than the above relation predicts.

For small x, $\epsilon \simeq \delta$ and there is no amplification of errors.

For large x, $\epsilon \simeq x\delta$ and we have strong amplification of the relative error. Eventually K_1, which is asymptotically given by $\frac{e^{-x}}{\sqrt{x}}$, becomes so small that it cannot be calculated without underflow and hence the function will return zero. Note that for large x the errors will be dominated by those of the standard function exp.

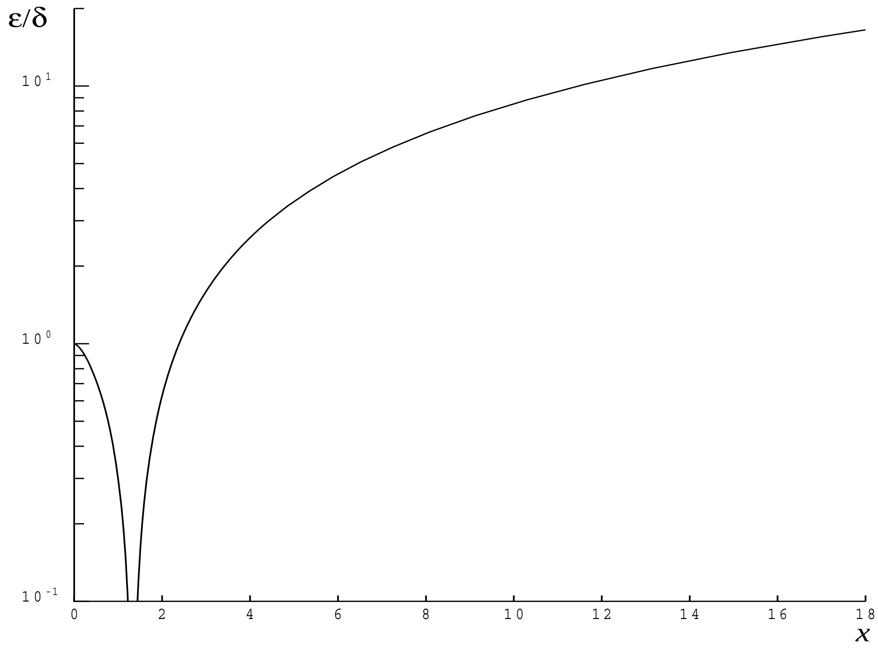


Figure 1

8 Further Comments

None.

9 Example

This example reads values of the argument x from a file, evaluates the function at each value of x and prints the results.

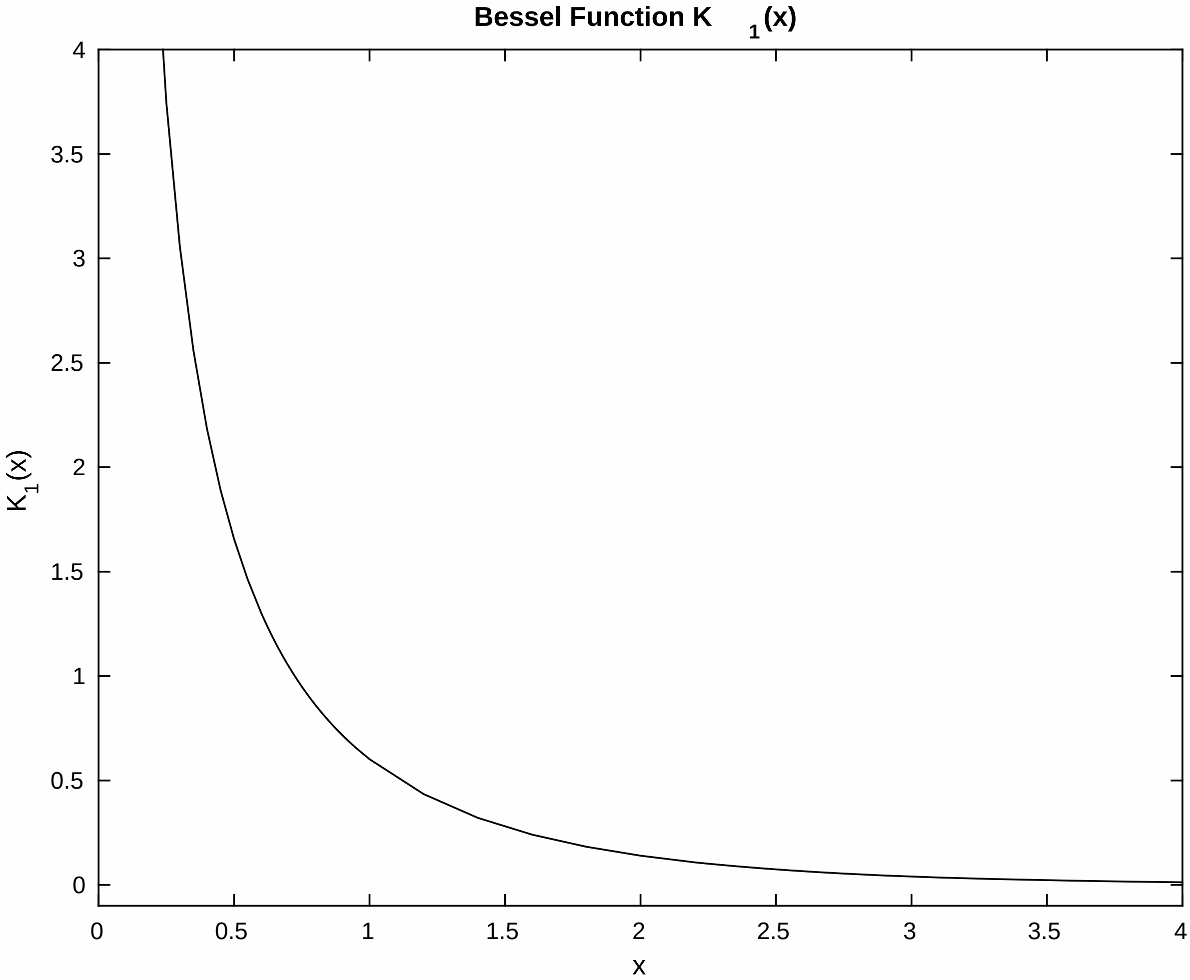
9.1 Program Text

```
     function s18ad_example

fprintf('s18ad example results\n\n');

x = [0.4   0.6   1.4   1.6    2.5    3.5    6    8    10   1000];
n = size(x,2);
result = x;

for j=1:n
  [result(j), ifail] = s18ad(x(j));
end

disp('      x         K_1(x)');
fprintf('%12.3e%12.3e\n',[x; result]);

s18ad_plot;
```

```
function s18ad_plot
  x = [0.2:0.05:1,1.2:0.2:4];
  for j = 1:numel(x)
    [K(j), ifail] = s18ad(x(j));
  end

  fig1 = figure;
  plot(x,K,'-r');
  xlabel('x');
  ylabel('K_1(x)');
  title('Bessel Function K_1(x)');
  axis([0 4 -0.1 4]);

  % print(fig1,'-dpng','-r75','s18ad_fig1.png');
  % print(fig1,'-deps','-r75','s18ad_fig1.eps');
```

9.2 Program Results

```
 s18ad example results

    x          K_1(x)
  4.000e-01   2.184e+00
  6.000e-01   1.303e+00
  1.400e+00   3.208e-01
  1.600e+00   2.406e-01
  2.500e+00   7.389e-02
  3.500e+00   2.224e-02
  6.000e+00   1.344e-03
  8.000e+00   1.554e-04
  1.000e+01   1.865e-05
  1.000e+03   0.000e+00
```


Bessel Function K 1(x)
K1(x)
x
0
0.5
1
1.5
2
2.5
3
3.5
4
0
0.5
1
1.5
2
2.5
3
3.5
4