NAG Toolbox for MATLAB

s18af

1 Purpose

s18af returns a value for the modified Bessel Function $I_1(x)$, via the function name.

2 Syntax

```
[result, ifail] = s18af(x)
```

3 Description

s18af evaluates an approximation to the modified Bessel Function of the first kind $I_1(x)$.

Note: $I_1(-x) = -I_1(x)$, so the approximation need only consider $x \geq 0$.

The function is based on three Chebyshev expansions:

For $0 < x \leq 4$,

$$I_1(x) = x\sum_{r=0}{}' a_r T_r(t), \qquad \text{where } t = 2\left(\frac{x}{4}\right)^2 - 1;$$

For $4 < x \leq 12$,

$$I_1(x) = e^x\sum_{r=0}{}' b_r T_r(t), \qquad \text{where } t = \frac{x-8}{4};$$

For $x > 12$,

$$I_1(x) = \frac{e^x}{\sqrt{x}}\sum_{r=0}{}' c_r T_r(t), \qquad \text{where } t = 2\left(\frac{12}{x}\right) - 1.$$

For small x, $I_1(x) \simeq x$. This approximation is used when x is sufficiently small for the result to be correct to ***machine precision***.

For large x, the function must fail because $I_1(x)$ cannot be represented without overflow.

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 – double scalar**

The argument x of the function.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **result – double scalar**

The result of the function.

2: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

x is too large. On soft failure the function returns the approximate value of $I_1(x)$ at the nearest valid argument.

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:

$$\epsilon \simeq \left| \frac{x I_0(x) - I_1(x)}{I_1(x)} \right| \delta.$$

Figure 1 shows the behaviour of the error amplification factor

$$\left| \frac{x I_0(x) - I_1(x)}{I_1(x)} \right|.$$

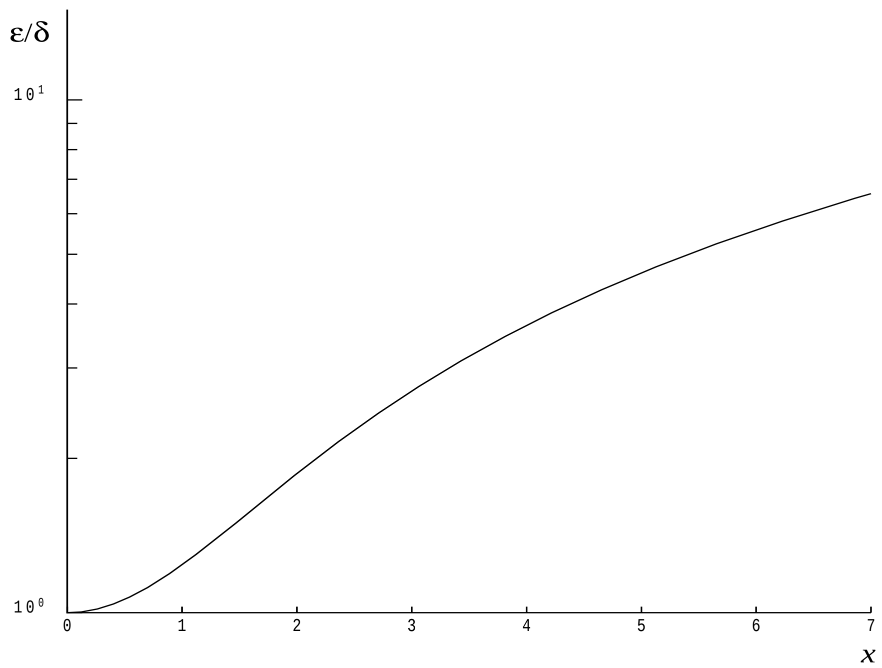

Figure 1

However, if δ is of the same order as ***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 errors. However the function must fail for quite moderate values of x because $I_1(x)$ would overflow; hence in practice the loss of accuracy for large x is not excessive. Note that for large x, the errors will be dominated by those of the standard function EXP.

8 Further Comments

None.

9 Example

```
x = 0;
[result, ifail] = s18af(x)
```

```
result =
     0
ifail =
           0
```