NAG Toolbox for MATLAB

s17aj

1 Purpose

s17aj returns a value of the derivative of the Airy function $\mathrm{Ai}(x)$, via the function name.

2 Syntax

```
[result, ifail] = s17aj(x)
```

3 Description

s17aj evaluates an approximation to the derivative of the Airy function $\mathrm{Ai}(x)$. It is based on a number of Chebyshev expansions.

For $x < -5$,

$$\mathrm{Ai}'(x) = \sqrt[4]{-x}\left[a(t)\cos z + \frac{b(t)}{\zeta}\sin z\right],$$

where $z = \frac{\pi}{4} + \zeta$, $\zeta = \frac{2}{3}\sqrt{-x^3}$ and $a(t)$ and $b(t)$ are expansions in variable $t = -2\left(\frac{5}{x}\right)^3 - 1$.

For $-5 \leq x \leq 0$,

$$\mathrm{Ai}'(x) = x^2 f(t) - g(t),$$

where f and g are expansions in $t = -2\left(\frac{x}{5}\right)^3 - 1$.

For $0 < x < 4.5$,

$$\mathrm{Ai}'(x) = e^{-11x/8} y(t),$$

where $y(t)$ is an expansion in $t = 4\left(\frac{x}{9}\right) - 1$.

For $4.5 \leq x < 9$,

$$\mathrm{Ai}'(x) = e^{-5x/2} v(t),$$

where $v(t)$ is an expansion in $t = 4\left(\frac{x}{9}\right) - 3$.

For $x \geq 9$,

$$\mathrm{Ai}'(x) = \sqrt[4]{-x} e^{-z} u(t),$$

where $z = \frac{2}{3}\sqrt{x^3}$ and $u(t)$ is an expansion in $t = 2\left(\frac{18}{z}\right) - 1$.

For $|x| <$ the square of the ***machine precision***, the result is set directly to $\mathrm{Ai}'(0)$. This both saves time and avoids possible intermediate underflows.

For large negative arguments, it becomes impossible to calculate a result for the oscillating function with any accuracy and so the function must fail. This occurs for $x < -\left(\frac{\sqrt{\pi}}{\epsilon}\right)^{4/7}$, where ϵ is the ***machine precision***.

For large positive arguments, where Ai' decays in an essentially exponential manner, there is a danger of underflow so the function must fail.

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 and positive. On soft failure, the function returns zero.

ifail $= 2$

x is too large and negative. On soft failure, the function returns zero.

7 Accuracy

For negative arguments the function is oscillatory and hence absolute error is the appropriate measure. In the positive region the function is essentially exponential in character and here relative error is needed. The absolute error, E, and the relative error, ϵ, are related in principle to the relative error in the argument, δ, by

$$E \simeq \left|x^2 \operatorname{Ai}(x)\right|\delta \qquad \epsilon \simeq \left|\frac{x^2 \operatorname{Ai}(x)}{\operatorname{Ai}'(x)}\right|\delta.$$

In practice, approximate equality is the best that can be expected. When δ, ϵ or E is of the order of the ***machine precision***, the errors in the result will be somewhat larger.

For small x, positive or negative, errors are strongly attenuated by the function and hence will be roughly bounded by the ***machine precision***.

For moderate to large negative x, the error, like the function, is oscillatory; however the amplitude of the error grows like

$$\frac{|x|^{7/4}}{\sqrt{\pi}}.$$

Therefore it becomes impossible to calculate the function with any accuracy if $|x|^{7/4} > \frac{\sqrt{\pi}}{\delta}$.

For large positive x, the relative error amplification is considerable:

$$\frac{\epsilon}{\delta} \simeq \sqrt{x^3}.$$

However, very large arguments are not possible due to the danger of underflow. Thus in practice error amplification is limited.

8 Further Comments

None.

9 Example

```
x = -10;
[result, ifail] = s17aj(x)
```

```
result =
    0.9963
ifail =
           0
```