NAG Toolbox

nag_fit_pade_app (e02ra)

1 Purpose

nag_fit_pade_app (e02ra) calculates the coefficients in a Padé approximant to a function from its user-supplied Maclaurin expansion.

2 Syntax

```
[a, b, ifail] = nag_fit_pade_app(ia, ib, c)
[a, b, ifail] = e02ra(ia, ib, c)
```

3 Description

Given a power series

$$c_0 + c_1 x + c_2 x^2 + \cdots + c_{l+m} x^{l+m} + \cdots$$

nag_fit_pade_app (e02ra) uses the coefficients c_i, for $i = 0, 1, \ldots, l + m$, to form the $[l/m]$ Padé approximant of the form

$$\frac{a_0 + a_1 x + a_2 x^2 + \cdots + a_l x^l}{b_0 + b_1 x + b_2 x^2 + \cdots + b_m x^m}$$

with b_0 defined to be unity. The two sets of coefficients a_j, for $j = 0, 1, \ldots, l$, and b_k, for $k = 0, 1, \ldots, m$, in the numerator and denominator are calculated by direct solution of the Padé equations (see Graves–Morris (1979)); these values are returned through the argument list unless the approximant is degenerate.

Padé approximation is a useful technique when values of a function are to be obtained from its Maclaurin expansion but convergence of the series is unacceptably slow or even nonexistent. It is based on the hypothesis of the existence of a sequence of convergent rational approximations, as described in Baker and Graves–Morris (1981) and Graves–Morris (1979).

Unless there are reasons to the contrary (as discussed in Chapter 4, Section 2, Chapters 5 and 6 of Baker and Graves–Morris (1981)), one normally uses the diagonal sequence of Padé approximants, namely

$$\{[m/m], m = 0, 1, 2, \ldots\}.$$

Subsequent evaluation of the approximant at a given value of x may be carried out using nag_fit_pade_eval (e02rb).

4 References

Baker G A Jr and Graves–Morris P R (1981) Padé approximants, Part 1: Basic theory *encyclopaedia of Mathematics and its Applications* Addison–Wesley

Graves–Morris P R (1979) The numerical calculation of Padé approximants *Padé Approximation and its Applications. Lecture Notes in Mathematics* (ed L Wuytack) **765** 231–245 Adison–Wesley

5 Parameters

5.1 Compulsory Input Parameters

1: **ia** – INTEGER
2: **ib** – INTEGER

ia must specify $l + 1$ and **ib** must specify $m + 1$, where l and m are the degrees of the numerator and denominator of the approximant, respectively.

Constraint: **ia** ≥ 1 and **ib** ≥ 1.

3: **c**(ic) – REAL (KIND=nag_wp) array

ic, the dimension of the array, must satisfy the constraint $ic \geq$ **ia** $+$ **ib** $- 1$.

c(i) must specify, for $i = 1, 2, \ldots, l + m + 1$, the coefficient of x^{i-1} in the given power series.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **a**(**ia**) – REAL (KIND=nag_wp) array

a$(j + 1)$, for $j = 1, 2, \ldots, l + 1$, contains the coefficient a_j in the numerator of the approximant.

2: **b**(**ib**) – REAL (KIND=nag_wp) array

b$(k + 1)$, for $k = 1, 2, \ldots, m + 1$, contains the coefficient b_k in the denominator of the approximant.

3: **ifail** – INTEGER

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, $jw <$ **ib** $\times (2 \times$ **ib** $+ 3)$,
or **ia** < 1,
or **ib** < 1,
or $ic <$ **ia** $+$ **ib** $- 1$

(so there are insufficient coefficients in the given power series to calculate the desired approximant).

ifail $= 2$

The Padé approximant is degenerate.

ifail $= -99$

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

ifail $= -399$

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

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

The solution should be the best possible to the extent to which the solution is determined by the input coefficients. It is recommended that you determine the locations of the zeros of the numerator and denominator polynomials, both to examine compatibility with the analytic structure of the given function and to detect defects. (Defects are nearby pole-zero pairs; defects close to $x = 0.0$ characterise ill-conditioning in the construction of the approximant.) Defects occur in regions where the approximation is necessarily inaccurate. The example program calls nag_zeros_poly_real (c02ag) to determine the above zeros.

It is easy to test the stability of the computed numerator and denominator coefficients by making small perturbations of the original Maclaurin series coefficients (e.g., c_l or c_{l+m}). These questions of intrinsic error of the approximants and computational error in their calculation are discussed in Chapter 2 of Baker and Graves–Morris (1981).

8 Further Comments

The time taken is approximately proportional to m^3.

9 Example

This example calculates the $[4/4]$ Padé approximant of e^x (whose power-series coefficients are first stored in the array **c**). The poles and zeros are then calculated to check the character of the $[4/4]$ Padé approximant.

9.1 Program Text

```
     function e02ra_example

fprintf('e02ra example results\n\n');

ia = nag_int(5);
ib = nag_int(5);
ic = ia + ib - 1;
c  = ones(ic,1);
for j = 3:ic
  c(j) = c(j-1)/double(j-1);
end

[a, b, ifail] = e02ra(ia, ib, c);

fprintf('The given series coefficients are\n');
fmt = '%13.4e%13.4e%13.4e%13.4e%13.4e\n';
fprintf(fmt,c);
fprintf('\n\nNumerator coefficients\n');
fprintf(fmt,a);
fprintf('\nDenominator coefficients\n');
fprintf(fmt,b);

% Calculate zeros of the approximant using c02ag.
dd(ia:-1:1) = a(1:ia);
[z, ifail] = c02ag(dd,ia-1);

fprintf('\nZeros of approximant are at\n\n');
cz = z(1,:) + i*z(2,:);
disp(cz');

% Calculate poles of the approximant using c02ag
dd(ib:-1:1) = b(1:ib);
```

```
[z, ifail] = c02ag(dd,ib-1);

fprintf('\nPoles of approximant are at\n\n');
cz = z(1,:) + i*z(2,:);
disp(cz');
```

9.2 Program Results

```
     e02ra example results

The given series coefficients are
   1.0000e+00   1.0000e+00   5.0000e-01   1.6667e-01   4.1667e-02
   8.3333e-03   1.3889e-03   1.9841e-04   2.4802e-05

Numerator coefficients
   1.0000e+00   5.0000e-01   1.0714e-01   1.1905e-02   5.9524e-04

Denominator coefficients
   1.0000e+00  -5.0000e-01   1.0714e-01  -1.1905e-02   5.9524e-04

Zeros of approximant are at

  -5.7924 - 1.7345i
  -5.7924 + 1.7345i
  -4.2076 - 5.3148i
  -4.2076 + 5.3148i

Poles of approximant are at

   5.7924 - 1.7345i
   5.7924 + 1.7345i
   4.2076 - 5.3148i
   4.2076 + 5.3148i
```