LATEX? :)

20/20

Cristian B. Jetomo
MATH 174 - B2L

Exercise 4.2: Improved Numerical Differentiation Formulas

In this exercise, we wish to derive a theoretical error term for the four-point backward formula given by

$$f'(x) \approx D(h) = \frac{f(x-3h) - f(x-2h) - 5f(x-h) + 5f(x)}{4h} \tag{1}$$

Note that by Taylor's theorem, the expansion for $f(x-h), f(x-2h)$, and $f(x-3h)$ are given as follows:

$$\begin{cases} f(x-h) &= f(x) - hf'(x) + \frac{h^2 f''(x)}{2!} - \frac{h^3 f'''(x)}{3!} + \cdots \\ f(x-2h) &= f(x) - 2hf'(x) + \frac{4h^2 f''(x)}{2!} - \frac{8h^3 f'''(x)}{3!} + \cdots \\ f(x-3h) &= f(x) - 3hf'(x) + \frac{9h^2 f''(x)}{2!} - \frac{27h^3 f'''(x)}{3!} + \cdots \end{cases}$$

This implies that

$$\begin{aligned} &-f(x-3h) + f(x-2h) + 5f(x-h) \\ &= (-1+1+5)\, f(x) - (-3+2+5)\, hf'(x) - (-27+8+5)\, \frac{h^3 f'''(x)}{3!} + \cdots \\ &= 5f(x) - 4hf'(x) + \frac{14h^3 f'''(x)}{6} + \cdots \end{aligned}$$

and consequently

$$\begin{aligned} 4hf'(x) &= f(x-3h) - f(x-2h) - 5f(x-h) + 5f(x) + \frac{7}{3}h^3 f'''(x) + \cdots \\ f'(x) &= \frac{f(x-3h) - f(x-2h) - 5f(x-h) + 5f(x)}{4h} + \frac{7}{12h}h^3 f'''(x) + \cdots \\ &= \frac{f(x-3h) - f(x-2h) - 5f(x-h) + 5f(x)}{4h} + \frac{7}{12}h^2 f'''(x) + \cdots \qquad (2) \\ &= \frac{f(x-3h) - f(x-2h) - 5f(x-h) + 5f(x)}{4h} + \frac{7}{12}h^2 f'''(\xi) \\ &= D(h) + \frac{7}{12}h^2 f'''(\xi) \end{aligned}$$

(2pts)

for some $\xi \in (x-3h, x)$. Note that since $x - 3h < \xi < x$, then as $h \to 0, \xi \to x$. If we let the exact value $D = f'(x)$, then

$$\lim_{h\to 0}\left|\frac{D - D(h)}{h^2}\right| = \lim_{h\to 0}\left|\frac{\frac{7}{12}h^2 f'''(\xi)}{h^2}\right| = \lim_{h\to 0}\left|\frac{7}{12}f'''(\xi)\right| = \lim_{h\to 0}\left|\frac{7}{12}f'''(x)\right|$$

which is a constant. This means that the theoretical error is $\frac{7}{12}h^2 f'''(\xi)$ and we can conclude that the order of the formula obtained in Equation 2 is $\mathcal{O}(h^2)$. This implies that the approximate $D(h)$ approaches to D as fast as h^2 approaches to 0. (3pts)

Now, suppose that $f(x) = e^x$ and we wish to estimate the exact value $f'(1) = e$ using Equation 2 and by substituting each of the step sizes $h = 10^{-i}, i = 0, 1, 2, 3, \ldots, 20$. After this, we then find the relative errors of the obtained estimates using the equation

$$\text{relative error} = \frac{|\text{exact} - \text{estimate}|}{|\text{exact}|}$$

Using MATLAB, we calculate the values required. The obtained values are tabulated and presented below.

Table 4.2.1. The estimates using the four-point backward formula for $D(h)$ and their corresponding relative errors for each value of h from 10^0 to 10^{-20}.

h	$D(h)$	relative errors (in %)
10^0	2.08971624609	2.31236355181 $\times 10^1$
10^{-1}	2.70401341372	5.24905644016 $\times 10^{-1}$
10^{-2}	2.71812494934	5.77126021942 $\times 10^{-3}$
10^{-3}	2.71828024449	5.82708972806 $\times 10^{-5}$
10^{-4}	2.71828181260	5.83367396192 $\times 10^{-7}$
10^{-5}	2.71828182825	7.64695266167 $\times 10^{-9}$
10^{-6}	2.71828182807	1.41818206128 $\times 10^{-8}$
10^{-7}	2.71828182630	7.95303367527 $\times 10^{-8}$
10^{-8}	2.71828182186	2.42901627103 $\times 10^{-7}$
10^{-9}	2.71828115572	2.47485951796 $\times 10^{-5}$
10^{-10}	2.71827449438	2.69805530704 $\times 10^{-4}$
10^{-11}	2.71822564457	2.06688972456 $\times 10^{-3}$
10^{-12}	2.71827005349	4.33176821054 $\times 10^{-4}$
10^{-13}	2.71782596428	1.67703058560 $\times 10^{-2}$
10^{-14}	2.70894418009	3.43512886556 $\times 10^{-1}$
10^{-15}	2.22044604925	1.83143548250 $\times 10^1$
10^{-16}	0.00000000000	1.00000000000 $\times 10^2$
10^{-17}	0.00000000000	1.00000000000 $\times 10^2$
10^{-18}	0.00000000000	1.00000000000 $\times 10^2$
10^{-19}	0.00000000000	1.00000000000 $\times 10^2$
10^{-20}	0.00000000000	1.00000000000 $\times 10^2$

(4pts)

As observe, the minimum relative error is obtained when $h = 10^{-5}$. Moreover, the relative errors of the estimates are decreasing from $h = 10^0$ to $h = 10^{-5}$. The magnitude of this decrease is by some factor of 10^2 which reflects the order (of accuracy) of the formula we obtained in Equation 2. We can also notice that from $h = 10^{-5}$ to $h = 10^{-20}$, the relative errors are increasing expect for some sudden decrease of its values at $h = 10^{-8}$ and $h = 10^{-11}$. This is expected due to machine precision incurred by the software used which truncates the actual values in the calculations. (1pt)

To resolve this error due to machine limitations, we shall improve our formula for $D = f'(x)$ using Richardson extrapolation. Note that as obtained in Equation 2,

$$D - D(h) = \frac{7}{12}h^2 f'''(\xi)$$

And since $\xi \to x$ as $h \to 0$, then

$$\begin{aligned}\lim_{h\to 0}\left|\frac{D - D(h) - \frac{7}{12}h^2 f'''(x)}{h^2}\right| &= \lim_{h\to 0}\left|\frac{\frac{7}{12}h^2 f'''(\xi) - \frac{7}{12}h^2 f'''(x)}{h^2}\right| \\ &= \lim_{h\to 0}\left|\frac{7}{12}\left[f'''(\xi) - f'''(x)\right]\right| \\ &= 0\end{aligned}$$

Therefore

$$D = D(h) + K_1 h^2 + o(h^2) \tag{3}$$

where $K_1 = \frac{7}{12} h^2 f'''(x)$. Varying the parameter h to $h/2$, we have

$$D = D(h/2) + K_1 (h^2/4) + o(h^2) \tag{4}$$

Subtracting corresponding sides of Equation 4 from Equation 3 we get

$$0 = D(h) - D(h/2) + \frac{3}{4} K_1 h^2 + o(h^2) \Longrightarrow K_1 h^2 = \frac{4}{3} [D(h/2) - D(h)] + o(h^2)$$

Substituting this to Equation 3, we obtain a new formula for the exact value D.

$$D = D(h) + \frac{4}{3} [D(h/2) - D(h)] + o(h^2) = \frac{4D(h/2) - D(h)}{3} + o(h^2) \tag{5}$$

So, a new approximating formula for D, which we will call the *extrapolated formula*, is given by

$$D \approx \frac{4D(h/2) - D(h)}{3} \tag{6}$$

(3pts)

Based on Equation 5, this extrapolated formula is of order $o(h^2)$ *(little oh)* which implies that the approximate obtained using Equation 6 approaches to the actual value D faster than h^2 approaches to 0. (2pts)

We now calculate the new estimates using Equation 6. Similarly with Table 4.2.1, we obtain the respective relative errors of the new estimates using the extrapolated formula. The result of the calculations are tabulated below.

Table 4.2.2. The respective relative errors (in %) of the estimates obtained using the four-point backward formula and the extrapolated formula for each value of h from 10^0 to 10^{-20}.

h	4-point backward formula	formula with Richardson extrapolation
10^0	$2.31236355181 \times 10^1$	$4.11896013303 \times 10^1$
10^{-1}	$5.24905644016 \times 10^{-1}$	$9.40752830927 \times 10^{-3}$
10^{-2}	$5.77126021942 \times 10^{-3}$	$1.03100442134 \times 10^{-5}$
10^{-3}	$5.82708972806 \times 10^{-5}$	$1.04079438057 \times 10^{-8}$
10^{-4}	$5.83367396192 \times 10^{-7}$	$8.94294443375 \times 10^{-10}$
10^{-5}	$7.64695266167 \times 10^{-9}$	$3.29037947329 \times 10^{-9}$
10^{-6}	$1.41818206128 \times 10^{-8}$	$1.41818206128 \times 10^{-8}$
10^{-7}	$7.95303367527 \times 10^{-8}$	$2.97358718440 \times 10^{-7}$
10^{-8}	$2.42901627103 \times 10^{-7}$	$2.42118549299 \times 10^{-6}$
10^{-9}	$2.47485951796 \times 10^{-5}$	$2.96575647171 \times 10^{-6}$
10^{-10}	$2.69805530704 \times 10^{-4}$	$4.87633917832 \times 10^{-4}$
10^{-11}	$2.06688972456 \times 10^{-3}$	$1.11394146757 \times 10^{-4}$
10^{-12}	$4.33176821054 \times 10^{-4}$	$2.13496618923 \times 10^{-2}$
10^{-13}	$1.67703058560 \times 10^{-2}$	$4.52427080122 \times 10^{-1}$
10^{-14}	$3.43512886556 \times 10^{-1}$	$6.87836450055 \times 10^0$
10^{-15}	$1.83143548250 \times 10^1$	$4.00971935383 \times 10^1$
10^{-16}	$1.00000000000 \times 10^2$	$1.00000000000 \times 10^2$
10^{-17}	$1.00000000000 \times 10^2$	$1.00000000000 \times 10^2$
10^{-18}	$1.00000000000 \times 10^2$	$1.00000000000 \times 10^2$
10^{-19}	$1.00000000000 \times 10^2$	$1.00000000000 \times 10^2$
10^{-20}	$1.00000000000 \times 10^2$	$1.00000000000 \times 10^2$

(4pts)

Notice from Table 4.2.2 in the previous page that the minimum relative error of the estimates obtained using the extrapolated formula is smaller than the one obtained using the four-point backward formula. In addition, this specifed error occured when $h = 10^{-4}$ as opposed to the minimum relative error of the four-point backward formula which occured when $h = 10^{-5}$. This reflects that the estimates obtained using the extrapolated formula converges faster to the actual value than the estimates obtained using the backward formula. However, it is still evident for both formulas that at some value of h, the machine precision eventually truncates the calculations and in return, changes the trend and increases the relative errors of our estimates. (1pt)

APPENDIX A

Program Used in MATLAB to construct Table 4.2.1 and Table 4.2.2

```
clc
clear
close all
format long

syms x
%% Initializing constants
a = 1;  % goal is to approximate f'(a)
numPts = 4;
numH = 21;

%% Constructing h and x values

% constructing uniform increments h
h = ones(1, numH);
h = 10*h;
for i = 1:numH
    h(i) = h(i)^(1-i);
end

% constructing x values using different increments h
X = zeros(numPts, numH);
X(1,:) = a*ones(1, numH);  % a = entries of 1st row of X
for j = 1:numH
    for i = 2:numPts
        X(i,j) = X(1,j) - (i-1)*h(j);
    end
end

%% f, estimates for f'(a) and relative errors
f = exp(X);  % functions values
exact = exp(1);  % exact e
D = zeros(1, numH);

% estimates D(h)
for j = 1:numH
    num = f(4,j) - f(3,j) - 5*f(2,j) + 5*f(1,j);
    den = 4*h(j);
    D(j) = num/den;  % approximating formula
end

disp('estimates D(h) = ')
disp(vpa(D, 6)')

% relative errors for D(h)
relError = zeros(1, numH);
for j = 1:numH
    numer = abs(exact - D(j));
    denom = abs(exact);
    relError(j) = numer/denom;
end

disp('rel Error (h) = ')
disp(vpa(relError*100, 6)')
```

```
%% solving new estimates using stepsize h/2
newh = h/2;

% constructing x values using h/2
newX = zeros(numPts, numH);
newX(1,:) = a*ones(1, numH);  % a = entries of 1st row of X
for j = 1:numH
    for i = 2:numPts
        newX(i,j) = newX(1,j) - (i-1)*newh(j);
    end
end

newf = exp(newX);
D_2 = zeros(1, numH);
% estimates D(h/2)
for j = 1:numH
    num = newf(4,j) - newf(3,j) - 5*newf(2,j) + 5*newf(1,j);
    den = 4*newh(j);
    D_2(j) = num/den;  % approximating formula
end


%% New Estimates using Richardson Extrapolation
% formula obtained:
% 4D(h/2) - D(h))/3
RE_est = zeros(1, numH);
for j = 1:numH
    RE_est(j) = (4*D_2(j) - D(j))/3;
end

disp('estimates Re_est = ')
disp(vpa(RE_est, 6)')

% relative errors for RE_est
newrelError = zeros(1, numH);
for j = 1:numH
    numer = abs(exact - RE_est(j));
    denom = abs(exact);
    newrelError(j) = numer/denom;
end

disp('rel Error (RE_est) = ')
disp(vpa(newrelError*100, 6)')
```