Exercise 9: Richardson Extrapolation and Romberg Integration
Exercise 9 - Numerical integration (II)

- Accurate numerical integration scheme
  - Richardson extrapolation
  - Romberg integration

- Implementation
  - Improve the accuracy of function evaluation (Pen & Paper)
  - Write a pseudocode for Romberg integration
  - Find velocity distribution in an air jet behind a turbofan engine (Coding)

!! NO Office hours on Thursday (10/05)
Richardson extrapolation

- **Taylor expansion**

\[
G(h) = G(0) + c_1 h + c_2 h^2 + \cdots , \quad (5.4.2)
\]

\[
G(h/2) = G + \frac{1}{2} c_1 h + \frac{1}{4} c_2 h^2 + \cdots . \quad (5.4.3)
\]

- **Reduce error by combination**

2\textsuperscript{nd} order accurate

\[
G_1(h) = 2 G(h/2) - G(h) = G + c'_2 h^2 + c'_3 h^3 + \cdots , \quad (5.4.4)
\]

3\textsuperscript{rd} order accurate

\[
G_2(h) = \frac{1}{3} \left( 4 G_1(h/2) - G_1(h) \right) = G + O(h^3) , \quad (5.4.5)
\]

- **Iterative scheme**

\[
G_n(h) = \frac{1}{2^n - 1} \left( 2^n G_{n-1}(h/2) - G_{n-1}(h) \right) = G + O(h^{n+1}) \quad (5.4.6)
\]

To evaluate this, we also need to compute \( G(h/4) \) in the first step.
Approximate an integral \( G = \int_{0}^{1} x^2 (49 - 25x^2) \, dx \) (exact value is 11.3)

\( E_k(h) \) is the error of \( k \)-th iteration of Romberg integration with step \( h \)

\[
E_0(0.5) = 0.0010 \\
E_0(0.25) = 0.0007
\]

First iteration gives \( E_1(0.5) = 0.0013 \) which is worse than both zero iterations
Sidenote: Order of Accuracy

- Q: Which of these methods approximates the quantity $Q$ better:

$$M_1(h) = Q + c_1 h^2 + c_2 h^3 + \ldots$$
$$M_2(h) = Q + d_2 h^3 + \ldots$$

- A: It depends!
  - for $h \to 0$, $M_2$ is better
  - For some fixed $h$, it depends on the constants

- High order of accuracy does not necessarily mean smaller absolute error (remember, there’s a constant in front of $h^2$ and $h^3$)

  - Let’s illustrate this using a convergence plot:

![Convergence plot](image)

  - In the region highlighted, the error for the 2nd order algorithm is smaller than that for the 3rd order algorithm
  - The order of accuracy just indicates how **fast** the error will drop as you refine the grid.
Romberg integration

- Recall the Romberg integration method for the approximation of

\[ I = \int_{a}^{b} f(x) \, dx \]

- We start with a set of trapezoidal approximations \( I_0^n \) for \( n = 1, 2, 4, 8, \ldots \):

\[
I_0^n = \frac{b - a}{2n} \left[ f(a) + f(b) + 2 \sum_{j=1}^{n-1} f(a + j \frac{b - a}{n}) \right]
\]

- Then we recursively calculate the higher order approximations according to the following expression:

\[
I_k^n = \frac{4^k I_k^{2n} - I_k^{n}}{4^k - 1}
\]

- You have to figure out how many initial integrations you need to obtain the desired order of accuracy
Computing initial integrals

- Many of the function evaluations are used in more than one refinement level (e.g. f(0), f(h), f(h/2)....)
- Don’t recompute them in every step! Instead, store and reuse them
Exercise 9

• Problem 1 (pen & paper)
  • **Aim:** compute function value accurately
  • Use Richardson extrapolation

• Problem 2 (pseudocode)
  • Write pseudocode for Romberg integration with the trapezoidal rule

• Problem 3 (engineering, coding)
  • **Aim:** compute the numerical integral with high precision using only a few function evaluations
  • Use Romberg integration
Problem 1: Accurate Function Evaluation

• Approximate the first derivative of a function $f$ by

$$f'(x) \approx \frac{f(x + h) - f(x)}{h} = G_0(h)$$

• Let $f(x) = x + e^x$, $x=0$, $h=0.4$

• Use Richardson extrapolation to compute $G_2(h)$

• Keep 5 decimal points in the calculation

• Compute the error for each evaluation (Exact value: $f'(0)=2$)
Problem 2

• Write pseudocode to compute an approximation to an integral using Romberg integration with the trapezoidal method.

• Fill the gaps in the skeleton code.

Algorithm 1 Romberg integration

Input:
function \( f(x) \)
interval boundaries \( a, b \)
number of iterations \( K \)

Output:
\( R^1_K = \text{integral}[K, 0] \) approximation to the integral \( \int_a^b f(x) \, dx \)

Steps:
maxNumIntervals \( \leftarrow 2^K \)

// Precompute and store function evaluations
hmin \( \leftarrow (b - a)/\text{maxNumIntervals} \)
for \( i \leftarrow 0, \ldots, \text{maxNumIntervals} \) do
    
end for

// Compute level 0 integrals
for \( r \leftarrow 0, \ldots, K \) do // refinement
    numIntervals \( \leftarrow 2^r \)
    step \( \leftarrow 2^{K-r} \) // step between two function evaluations for this refinement
    
end for

• Hint: Use this as a basis for Problem 3, where you have to implement Romberg integration!
Problem 3: Romberg Integration

- On the picture to the right you can see airplane contrails.
- The speed of the air in the jet has the following radial dependency:

\[ u(r) = u_{\text{exhaust}} + (u_{\text{freestream}} - u_{\text{exhaust}}) \cdot \text{erf} \left( \frac{2r}{R} \right) \]

- \( \text{erf} \) is the error function:

\[ \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_{0}^{x} e^{-t^2} \, dt \]

- It has no analytical expression.
Problem 3 - Romberg integration

• Compute the Romberg integration of erf(0.5)
  • Compute the approximation
  • Output all the intermediate integral values and corresponding errors
  • Observe the diminishing error with increased theoretical accuracy
  • Exact value is: erf(0.5) = 0.5204998778130467

• Compute the velocity profile in the jet and visualize it