Richardson Extrapolation

Richardson Extrapolation

✓ **Accelerated convergence of a sequence**

- Assume that an approximation converging to the value of a function **F(t)** as h → 0

$$\underbrace{F(t) \approx A(t,h)}_{\text{Approximation of function}} \Rightarrow F(t) = A(t,h) + \underbrace{\alpha_0(t)\,h^m + \alpha_1(t)\,h^{m+1}}_{\text{Leading error terms (depend on t)}} + \dots$$

- Examples:
  - **Differentiation**

Forward approximation

Leading error

$$f'(t) \approx A(t,h) = \frac{f(t+h) - f(t)}{h} \Rightarrow f'(t) = A(t,h) + \alpha_0 h + \alpha_1 h^2 + \dots$$

  - **Integration**

No. of subintervals

Forward rectangle approx.

Leading error

$$\underbrace{F(t) = \int_a^t f(x)dx}_{\text{Integral function}} \approx A(t,h) = \sum_{i=0}^{n=(t-a)/h} f(a+ih) \times h \Rightarrow \int_a^t f(x)dx = A(t,h) + \alpha_0(t)\,h + \alpha_1(t)\,h^2 + \dots$$

Richardson Extrapolation

✓ **We can use a clever trick to get a new approximation**

- Start with the original approximation

$$F(t) \approx A(t,h) \Rightarrow F(t) = A(t,h) + \alpha_0(t)\, h^m + \alpha_1(t)\, h^{m+1} + \dots$$

- Substitute **h/2** into the original approximation and eliminate the leading error term

$$F(t) \approx A(t,h/2) \Rightarrow 2^m\, F(t) = 2^m\, A(t,h/2) + \alpha_0(t)\, h^m + \frac{\alpha_1(t)}{2}\, h^{m+1} + \dots$$

- Now subtract the two approximations
  - o Notice that the leading error term cancels

New leading error (order = m+1)

$$(2^m - 1)F(t) = (2^m\, A(t,h/2) - A(t,h)) - \frac{\alpha_1(t)}{2} h^{m+1} + \dots$$

- Construct a new approximation with improved leading error term
  - Repeat the algorithm for the designed order of error h^n (going from $\mathbf{h^m}$ to $\mathbf{h^n}$)

$A_k \rightarrow h^{m+k}$

$$F(t) \approx \frac{2^m\, A(t,\frac{h}{2}) - A(t,h)}{2^m - 1} \Rightarrow \boxed{A_{k+1}(t,h) = \frac{2^{m+k} A_k(t,\frac{h}{2}) - A_k(t,h)}{2^{m+k} - 1}}$$

Richardson's extrapolation iterations

Richardson Extrapolation

✓ **How to estimate leading term's order (m)**

- We can use the starting formula and use two cases: h/2, h/3

$$A(t, \frac{h}{2}) \approx A(t, \frac{h}{3}) \Rightarrow$$ Solve for **m**

Solve this to find "m"

$$\frac{2^m\, A(t, \frac{h}{2}) - A(t,h)}{2^m - 1} \approx \frac{3^m\, A(t, \frac{h}{3}) - A(t,h)}{3^m - 1} \Rightarrow \boxed{A(t, \frac{h}{2}) + \frac{A(t, \frac{h}{2}) - A(t,h)}{2^m - 1} \approx A(t, \frac{h}{3}) + \frac{A(t, \frac{h}{3}) - A(t,h)}{3^m - 1}}$$

- What is the meaning of accelerated convergence?
  - Consider sequence $\{a_n\}$ that converges to **A**
  - Assume that we can construct from a_n a new sequence $\{b_n\}$ that converges to **A** and has the property: $$\lim_{n\to\infty} \frac{b_n - A}{a_n - A} = 0$$
  - We say that b_n has a faster convergence than a_n

Richardson Extrapolation

✓ **Implementing in java**

- Define Richardson function **A(t, h)** as an interface
  - Functional interface
  - RichardsonFunction: (var1, var2) -> some value
- Define a class that implements the Richardson extrapolation

Richardson extrapolation

From Wikipedia, the free encyclopedia

In numerical analysis, Richardson extrapolation is a sequence acceleration method, used to improve the rate of convergence of a sequence. It is named after Lewis Fry Richardson, who introduced the technique in the early 20th century. In the words of Birkhoff and Rota, "its usefulness for practical computations can hardly be overestimated."

Practical applications of Richardson extrapolation include Romberg integration, which applies Richardson extrapolation to the **trapezoid rule**, and the Bulirsch–Stoer algorithm for solving ordinary differential equations.