Richardson Extrapolation

Recall, with two expressions of derivatives

$\nabla f^{O(h)}$ & $\Delta f^{O(h)}$ & their associated errors.

we can combine them to form a

new estimate with leading errors eliminated!

$$\delta f(x) = \nabla f - \Delta f \quad O(h^2)$$

Richardson Extrapolation

— in a similar spirit is to achieve the same goal with a single expression —

☆ (the idea is general wherever the error of an expression is known)

example with derivative

Central difference

① with h: $$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{6} f'''(x) + O(h^4)$$

Using a diff step size (but with a similar expression)

$\underline{h \to 2h}$, we can get a diff approx to $f'(x)$!

② with $2h$: $$f'(x) = \frac{f(x+2h) - f(x-2h)}{4h} - \frac{(2h)^2}{6} f'''(x) + O(h^4)$$

☆ By doubling the step size, the error is $\underline{\underline{4x}}$! ($h^2$)

With these two expressions, we can get a better approx by eliminating the h^2 term!

$\rightarrow$ ① $- \frac{②}{4}$ & solve for $f'(x)$:

$$f'(x) - \frac{f'(x)}{4} = \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{16h}$$

$$\cancel{-\frac{h^2}{6} f'''(x)} + \cancel{\frac{1}{4}\frac{(4h^2)}{6}} f'''(x) + O(h^4)$$

$$f'(x) = \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h} + O(h^4)$$

$$\left(\cancel{16}^{4} \times \frac{3}{\cancel{4}} = 12\right)$$

$\Rightarrow$ this expression for $f'(x)$ is accurate to $O(h^4)$!

---

check with next result:

$$D_2(h) = \frac{4D_1(h) - D_1(2h)}{3}$$

$$= \frac{4\left(\frac{f(h^+) - f(h^-)}{2h}\right) - \left(\frac{f(2h^+) - f(2h^-)}{4h}\right)}{3}$$

$$= \frac{f(2h^-) - 8f(h^-) + 8f(h^+) - f(2h^+)}{12h}$$

This procedure can be implemented (3)

numerically in a ~~recursive~~ iterative way :

(1) Let say $D_1(h)$ is our numerically

Computed result (diff, integration, etc.)
with step size h.

& we know its error to be of order $\underline{h^2}$.

(2) We can get a better estimate by

(a) Calculating $D_1(2h)$ [the same expression with $2h$]

Since $\Delta \sim h^2$, we know that

$D_1(2h)$ will have $4\times$ errors of $D_1(h)$.

(b) Calculate the error in $D_1(h)$ by :

$$\Delta = \frac{D_1(2h) - D_1(h)}{2^2 - 1 \ (=3)} \quad \leftarrow \text{diff} = 3\times\text{error in } D_1$$

a better estimate

(c) $$\boxed{\underline{D_2(h)} = D_1(h) + \Delta = D_1(h) - \frac{D_1(2h) - D_1(h)}{2^2 - 1}}$$

$\Rightarrow$ $D_2(h)$ will be of $\mathcal{O}(h^4)$!

(3) We can continue with the knowledge that $D_2(h)$ is of $O(h^4)$.

So, $D_2(2h)$ will have $2^4 \times$ the error of $D_2(h)$.

A better estimate will be

$$\boxed{D_3(h) = D_2(h) - \frac{D_2(2h) - D_2(h)}{2^4 - 1}}$$

$$D_{i+1}(h) = D_i(h) - \frac{D_i(2h) - D_i(h)}{2^{2i} - 1}$$
$$= \frac{4^i D_i(h) - D_i(2h)}{4^i - 1}$$

Romberg Integration

- Since we know the error estimate for the trapezoid rule to be $O(h^2)$, we can use Richardson extrapolation to improve our result.

Trapezoid rule :

(5)

$T_{m,0}$ (result for $N = 2^m N_0$)

$$h = \frac{(b-a)}{N}$$

$T_{m+1,0}$ (result with $N = 2^{m+1} N_0$)

intervals halfed!

$\uparrow$ # of N $\quad \uparrow$ level of extrapolation

Then,
$$T_{m+1,1} = T_{m+1,0} - \left[\frac{T_{m,0} - T_{m+1,0}}{2^2 - 1}\right]$$

$$= \frac{4\,T_{m+1,0} - T_{m,0}}{3}$$

In general, we have the ~~recursive~~ iterative scheme :

$$\boxed{T_{m+k,k} = \frac{4^k\,T_{m+k,k-1} - T_{m+k-1,k-1}}{4^k - 1}}$$

Romberg's Scheme

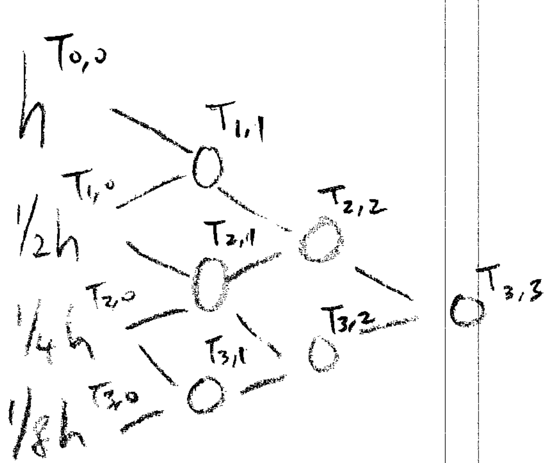
$T_{m,k}$

$\longrightarrow$ k (order of extrapolation)

m $\downarrow$ (step size)

$N = 2^m N_0$

$h = \frac{(b-a)}{N}$

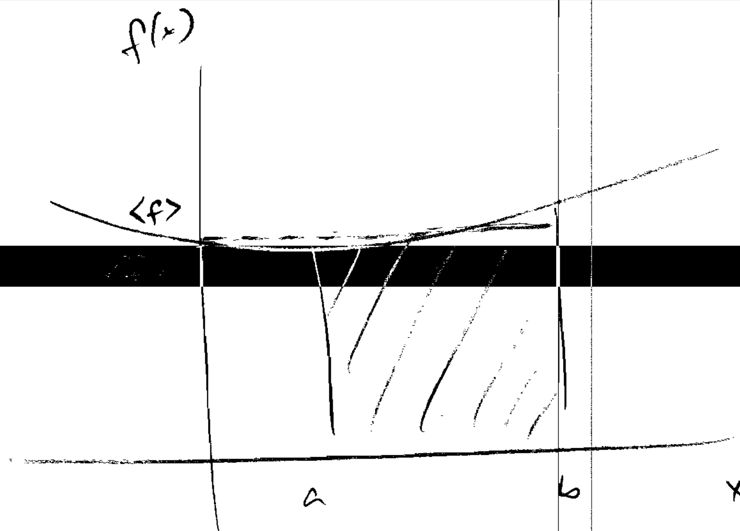


~ tree in Neville's Scheme

going down this column is what you would do to improve accuracy <u>without</u> extrapolation.

Monte Carlo Integration

– Not based on approx. integrant with polynomials.

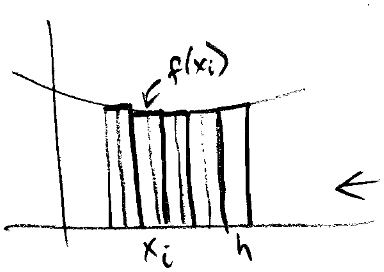


$$\int_a^b f(x)\,dx = \text{area under curve.}$$

$$\rightarrow \quad I \sim \sum f(x_i)\,h = \sum f(x_i)\,\frac{(b-a)}{N}$$

$$= \underset{\text{width}}{(b-a)}\ \langle f \rangle$$

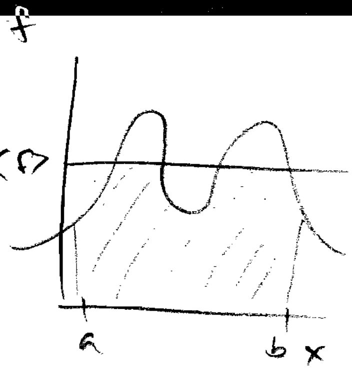
$\langle f \rangle$: average of f on grid pts



← simplest approx. (rect rule)

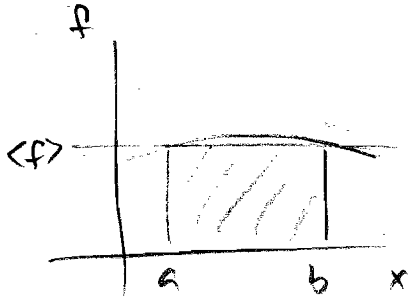
<u>Note</u> : with a fixed number of N,

<u>Case 1</u> :



$$I^1_{MC} = (b-a)\langle f\rangle$$

<u>Case 2</u> :



$$I^2_{MC} = (b-a)\langle f\rangle$$

☆ I^2_{MC} will be more accurate than I^1_{MC} (for a fixed N)

$\uparrow$ f is ~ uniform (constant)

$\uparrow$ f varies more in $[a,b]$

$\Rightarrow$ In general, if $f(x)$ ~ constant in $[a,b]$, the I_{MC} will converge faster with less N !

Enhanced Monte Carlo (Metropolis' Importance sampling)

- let say we can find $g(x)$ [later]

such that $g(x) \simeq f(x)$ in $[a,b]$

And, we can evalute the integral of $g(x)$ easily :

$$Y = h(x) = \int^{x} g(x')\,dx'$$

then we have,

$$\int_a^b f(x)\,dx = \int_a^b \frac{f(x)}{g(x)}\, g(x)\,dx$$

change variable $h(x) = Y$.

$$= \int_{h(a)}^{h(b)} \frac{f(h^{-1}(Y))}{g(h^{-1}(Y))}\,dY$$

this will be nearly uniform.

☆ In terms of this new variable $Y = \int^{x} g(x)\,dx$, the integral will have a better convergence characteristics!

Note: to stop → look at relative errors

$$\left|\frac{I_{N'} - I_N}{I_N}\right| < \varepsilon \qquad N', N \text{ different set "N" \# of pts.}$$

☆

— MC integration is a random process!

— each realization of a sequence of random #
will give slightly different I_N!

e.g. $\int_0^1 e^x \, dx$ with $N = 100$, $N = 400$
(distribution of 10,000 runs)

Fig. 4.12 in DeVries

Also, http://complex.gmu.edu/www-phys/phys510/notes/readme.html

(programs & graphs)