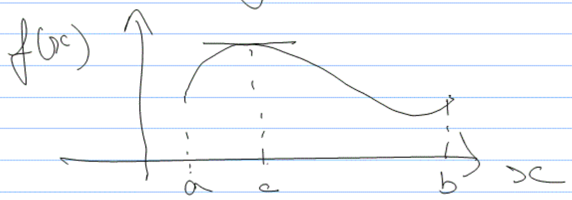
Note Title 8/6/2013

Numerical Methods – SE 288.

Rolle's Thorem:

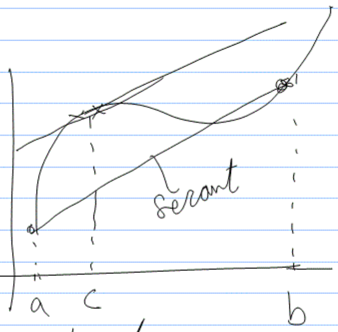
suppose $f \in C[a,b]$ and f is differentiable (a,b). If $f(a) = f(b)$ then there exists $c \in (a,b)$ with $f'(c) = 0$



Mean value thorem:

If $f \in C[a,b]$ and f is diff on (a,b) then a number $c \in (a,b)$ exists with

$$f'(c) = \frac{f(b) - f(a)}{b-a}$$



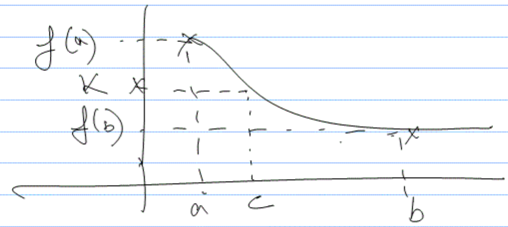
Remark:

Given arc between two end points (a & b), There is atleast one point at which the arc is parallel to the secant through its endpoints.

Intermediate value theorem:

If $f \in C[a,b]$ and K is any number between $f(a)$ and $f(b)$, Then, there exist $c \in (a,b)$ for which

$$f(c) = K$$

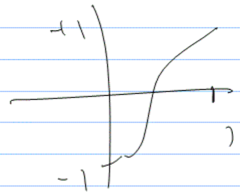


Ex: Show that $x^5 - 2x^3 + 3x^2 - 1 = 0$ has a Solution in the interval $[0,1]$

Let $f(x) = x^5 - 2x^3 + 3x^2 - 1$

$f(0) = -1 < 0$

$f(1) = 1 > 0$



By IMVT

$\Rightarrow$ $f(x)$ is having atleast one root in $[0,1]$

Taylor's Theorem:

Any function which satisfies certain conditions can be expressed as Taylor series.

Statement: Suppose $f \in C^n[a,b]$ and f^{n+1} exists on $[a,b]$ and $x_0 \in [a,b]$. Then, for every $x \in [a,b]$ there exists a number $\xi(x)$ between x_0 and x with

$$f(x) = P_n(x) + R_n(x) \quad , \quad \text{where}$$

$$P_n(x) = f(x_0) + f'(x_0)\,(x-x_0) + \cdots + \frac{f^n(x_0)}{n!}(x-x_0)^n$$

$$R_n(x) = \frac{f^{n+1}(\xi(x))}{(n+1)!}\,(x-x_0)^{n+1}.$$

Here, $P_n(x)$ is the n^{th} Taylor polynomial

$R_n(x)$ is called the truncation error associated with $P_n(x)$

Remarks:

1. $n \to \infty$ in $P_n(x) \Rightarrow$ Taylor Series of f about x_0

2. If $x_0 = 0 \Rightarrow$ Maclaurin Polynomial & Series (as $n \to \infty$).

Ex: Determine the second Taylor polynomial of $f(x) = \cos x$ about $x_0 = 0$, and use this polynomial to approximate $\cos(0.01)$.

$f(x) = \cos x \qquad f \in C^\infty[\mathbb{R}]$

$\Rightarrow$ Taylor theorem can be applied

$f(x) = \cos x \qquad f'(x) = -\sin x, \quad f''(x) = -\cos x$

$f'''(x) = \sin x.$

$f(x)$ about $x_0 = 0. \Rightarrow f(x_0) = f(0) = 1, \quad f'(0) = 0$

$f''(0) = -1$

For $n = 2$

$$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(\xi(x))}{3!}x^3$$

$$f(x) = 1 + 0 - \frac{1}{2}x^2 + \frac{1}{6}\sin(\xi(x))\,x^3$$

Here, $\xi(x)$ is some (unknown) number between x_0 & x

when $x = 0.01$

$$f(0.01) = \cos(0.01) = 1 - \frac{1}{2}(0.01)^2 + \frac{1}{6}\sin(\xi(0.01))\,(0.01)^3$$

$$f(0.01) = 0.99995 + \frac{10^{-6}}{6}\sin(\xi(0.01))$$

$\Rightarrow$ The approximation to $\cos(0.01)$ given by the Taylor polynomial is 0.99995

The Truncation error is $\boxed{\frac{10^{-6}}{6}\sin(\xi(0.01))}$

Exercise:

(1) Use Second Taylor Polynomial and its reminder term to approximate $\int_0^{0.1}\cos x\,dx$ about $x_0 = 0$: Hint use 2nd taylor polynomial in the integral

Round-off error.

The error which occurs while performing real number calculations (in computers)

Eg:

- $(\sqrt[3]{3})^3 \neq 3$ in Comp & Calculators

each real-number is represented by only a fixed (or) finite number of digits.

Representation of long real in 64-bit machine

- first bit is a sign indicator (s)
- followed by 11-bit exponent (c) characteristic with base 2 $[2^{11}-1 = 2047]$
- followed by 52 bit binary fraction (f) [mantisa]

Using these, a floating point is represented by

$$(-1)^s \, 2^{c-1023} (1+f)$$

Eg: 0 10000000011 101110010001000 0

$\uparrow_1$ $\uparrow_2$ $\uparrow_{12}$ $\uparrow_{13}$ $\uparrow_{64}$

$$S = 0$$

$$C = 1*2^{10} + 0*2^{9} + \cdots + 0*2^{2} + 1\times 2^{1} + 1*2^{0}$$

$$= 1024 + 2 + 1 = 1027$$

$$f = 1\left(\tfrac{1}{2}\right)^1 + 0\left(\tfrac{1}{2}\right)^2 + 1\left(\tfrac{1}{2}\right)^3 + 1\cdot\left(\tfrac{1}{2}\right)^4 + 1\left(\tfrac{1}{2}\right)^5 + 0\left(\tfrac{1}{2}\right)^6 + 0\left(\tfrac{1}{2}\right)^7$$

$$+ 1*\left(\tfrac{1}{2}\right)^8 + 0\left(\tfrac{1}{2}\right)^9 + \cdots + 0(\ )$$

$$(-1)^S\, 2^{C-1023}(1+f) = (-1)^0 * 2^{1027-1023}\left(1 + \frac{1}{2} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{256} + \frac{1}{4096}\right)$$

$$= 16\left(1 + \left(\frac{1}{2} + \frac{1}{8} \quad \cdots \right)\right)$$

$$= 27.56640625$$

<u>Remark:</u>

– Smallest positive number that can be represented by the above representation has

$$S = 0,\quad C = 1,\quad f = 0$$

$$\Rightarrow 2^{-1022}(1+0) \simeq 0.2225\times 10^{-307}$$

and the largest has $S = 0$, $C = 2046$ & $f = 1 - 2^{-52}$

$$\Rightarrow 2^{1023}\left(2 - 2^{-52}\right) \simeq 0.17977 \times 10^{309}$$

- In Calculations, if a number occurs less than 2^{-1022} results in underflow.
  and greater than $2^{1023}\left(2-2^{-52}\right)$ results in overflow.

Type of errors;

Suppose p^* is an approximation of P

The absolute error $|p^* - p|$

the relative error $\dfrac{|p - p^*|}{|p|}$

Eg:

If $p = 0.3 \times 10^1$ & $p^* = 0.31 \times 10^1$

absolute error $= 0.1$

relative error $= 0.33\bar{3}3 \times 10^{-1}$ repetation of the last digit

If $p = 0.3 \times 10^4$ & $p^* = 0.31 \times 10^4$

absolute error $= 0.1 \times 10^{3}$

relative error $= 0.333\bar{3} \times 10^{-1}$

Remark:

- absolute error can be misleading
- relative error is more meaningful, since the size of the value is taken into consideration.

Norm:

Let $x \in \mathbb{R}^N$, N- dimension, then.

$$x = (x_1, x_2, \ldots, x_N)^T$$

$$\|x\|_\infty = \max |x_i| \qquad 1 \leq i \leq N$$

$$\|x\|_2 = \left(\sum_{i=1}^{N} x_i^2 \right)^{1/2}$$

Eg: Let $x = (2, 3)$

$$\|x\|_2 = \left[2^2 + 3^2 \right]^{1/2} = \sqrt{13}$$

Properties of Norm:

(i) $\|x\| \geq 0$ & $\|x\| = 0$ iff $x = 0$

(ii) For any $\lambda \in \mathbb{R}$

$$\|\lambda x\| = |\lambda| \, \|x\|$$

<u>Other relations</u>

For any $x, y \in \mathbb{R}^N$

(i) $|x \cdot y| \leq \|x\| \, \|y\|$

(ii) $\|x + y\| \leq \|x\| + \|y\|$

Math in Computers

Let A, B & C are floating point numbers

$$(A+B)+C \underset{\text{in Computers}}{\neq} A + (B+C)$$

$$(A+B)+C = fl\left(fl\left(fl(A) + fl(B)\right) + fl(c)\right)$$

- leads to errors called "Round-off" error
- unavoidable
- generally, as no of operations increases so does roundoff error.

Q: Is it manageable.

<u>Class of error growth:</u>

Let $\varepsilon_0 > 0$ be the initial error and ε_N be the error at Nth iteration/operation

(a) $\varepsilon_N \propto N c \varepsilon_0$ (linear growth, generally ok)

(b) $\varepsilon_N \propto C^N \varepsilon_0$ (exponential growth, C is independent of N)

Let us consider the following two examples:

(a) $P_n = \frac{1}{3} P_{n-1}$ (b) $P_n = \frac{10}{3} P_{n-1} - P_{n-2}$

(a) <u>Case 0:</u> Assume that constants are represented exactly.

i.e. $\frac{1}{3} = 0.333 \ldots$

$$P_n = \frac{1}{3} P_{n-1} \quad \text{---} (1)$$

$$P_n^* = \frac{1}{3} P_{n-1}^* \quad \text{---} (2)$$

① − ② $\quad |P_n - P_n^*| = \frac{1}{3} |P_{n-1} - P_{n-1}^*|$

$$\varepsilon_n = \frac{1}{3} \varepsilon_{n-1} \quad \text{—— ③}$$

③ ⟹ $\quad \varepsilon_1 = \frac{1}{3}\varepsilon_0 \qquad (n=1)$

$\varepsilon_2 = \frac{1}{3}\varepsilon_1 \qquad (n=2)$

$= \left(\frac{1}{3}\right)^2 \varepsilon_0 \qquad \vdots$

$$\varepsilon_n = \left(\frac{1}{3}\right)^n \varepsilon_0$$

But here $\displaystyle \lim_{n\to\infty} \varepsilon_n = \varepsilon_0 \lim_{n\to\infty} (1/3)^n = 0$

(a) Case 1 $\quad$ if the constants are not represented exactly.

i.e. $\quad \frac{1}{3} = \hat{\frac{1}{3}} + \delta \quad$ where δ represents round-off error.

$\hat{\ }$ ← truncated value

$P_n = \frac{1}{3} P_{n-1}$ —— ① $\qquad$ (exact)

$P_n^* = \left(\hat{\frac{1}{3}}\right) P_{n-1}^*$ $\qquad$ (approximate)

$= \left(\frac{1}{3} - \delta\right) P_{n-1}^*$ —— ②

(1) − (2) $\quad p_n - p_n^* = \frac{1}{3}\left(p_{n-1} - p_{n-1}^*\right) + \delta\, p_{n-1}^*$

$$\varepsilon_n = \frac{1}{3}\varepsilon_{n-1} + \delta\, p_{n-1}^*$$

$$\Rightarrow \quad \varepsilon_1 = \frac{1}{3}\varepsilon_0 + \delta\, p_0^*$$

$$\varepsilon_2 = \frac{1}{3}\varepsilon_1 + \delta\, p_1^*$$

$$= \frac{1}{3}\left(\frac{1}{3}\varepsilon_0 + \delta\, p_0^*\right) + \delta\, p_1^*$$

$$\varepsilon_2 = \left(\frac{1}{3}\right)^2 \varepsilon_0 + \frac{1}{3}\delta\, p_0^* + \delta\, p_1^*$$

$$\vdots$$

$$\varepsilon_n = \left(\frac{1}{3}\right)^n \varepsilon_0 + \underbrace{\left(\frac{1}{3}\right)^{n-1}\delta\, p_0^*} + \left(\frac{1}{3}\right)^{n-2}\delta\, p_1^* + \cdots + \delta\, p_{n-1}^*$$

$$\Rightarrow \quad \varepsilon_n \le \left(\frac{1}{3}\right)^n \varepsilon_0 + n\,\delta\, p_0^*$$

⇒ errors, once introduced, it grows linearly but bounded

(b)

(b)
$$P_n = \frac{10}{3} P_{n-1} - P_{n-2}$$
$$P_n^* = \frac{10}{3} P_{n-1}^* - P_{n-2}^*$$

$$\varepsilon_n = \frac{10}{3} \varepsilon_{n-1} - \varepsilon_{n-2}$$

$$\Rightarrow \quad \varepsilon_n - \frac{10}{3} \varepsilon_{n-1} + \varepsilon_{n-2} = 0$$

Let $\varepsilon_n \propto \lambda^n$ ie $\varepsilon_n = c\,\lambda^n$

$$\Rightarrow \quad \lambda^n - \frac{10}{3} \lambda^{n-1} + \lambda^{n-2} = 0$$

$\div$ by λ^{n-2} to get

$$\lambda^2 - \frac{10}{3} \lambda + 1 = 0$$

$$\Rightarrow \quad \lambda_1 = \frac{1}{3} \;\&\; \lambda_2 = 3$$

ie $\varepsilon_n = c_1 (1/3)^n + c_2 (3)^n$

$$\lim_{n\to\infty} \varepsilon_n = c_1 \underbrace{\lim_{n\to\infty} (1/3)^n}_{\to 0} + c_2 \underbrace{\lim_{n\to\infty} 3^n}_{\to \infty}$$

$\Rightarrow$ once the error induced, they grow exponentially.

Solution of nonlinear equation with one variable

Ex: Find the root of $f(x) = 3x - e^{x}$ in $[1, 2]$

Bisection Method:

Suppose $f \in C[a,b]$ with $f(a)$ and $f(b)$ are of opposite sign. By IMVT there exists a value $p \in (a,b)$ such that $f(p) = 0$

$\Rightarrow$ p is the zero of the function "$f(x)$".

Method:

Set $a_1 = a, \quad b_1 = b$

Let $p_1 = \dfrac{a_1 + b_1}{2}$ (mid point)

If $f(p_1) = 0 \Rightarrow p = p_1$

If $f(p_1) \neq 0$ if $f(p_1) f(a_1) < 0 \Rightarrow p \in (a_1, p_1)$

Set $a_2 = a_1, \; b_2 = p_1$

else if $f(p_1) f(b_1) < 0 \Rightarrow p \in (p_1, b_1)$

Set $a_2 = p_1$ & $b_2 = b_1$

$\vdots$

$p_2 = \dfrac{a_2 + b_2}{2}$

$\vdots$

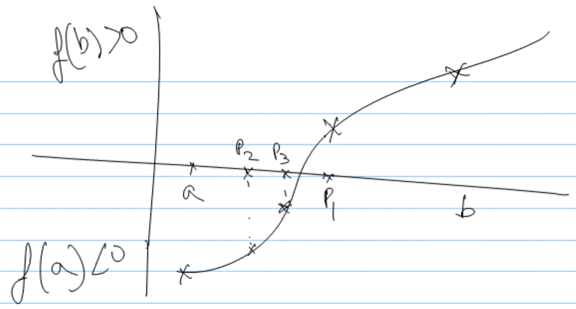
$p_n =$

(i) $|f(p_n)| < \varepsilon$ ($f(p_n)$ may be close to zero but p_n may not be close to p)

(ii) $|p_n - p_{n-1}| < \varepsilon$ ($|p_n - p_{n-1}|$ converges but the sequence may diverge $p_n = \sum_{k=1}^{n} \frac{1}{k}$)

(iii) $\dfrac{|f(p_n) - f(p_{n-1})|}{|f(p_n)|} < \varepsilon$ (Preferable)

Ex: for case (i) $f(x) = (x-1)^{10}$



Also, fix the no of iteration in computers

Ex:

(i) $f(x) = \sqrt{x} - \cos x$ in $[0,1]$

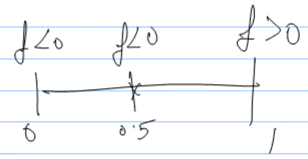
(ii) $f(x) = 3x - e^x$ in $[1,2]$

Using Bisection method find the solution

Case (1) $f(x) = \sqrt{x} - \cos x$

$f(0) < 0$

$f(1) > 0$



$P_1 = 0.5$, $f(0.5) < 0$

$P_2 = 0.75$

Fixed point Iteration.

A number "p" is a fixed point for a given function $f(x)$ if $f(p) = p$.

Application.

Find a root "p" of a function $f(x)$ for which $f(p) = 0$

We can define a function "$g(x)$" with a fixed point at p in many ways. For example

(i) $g(x) = x - f(x)$

Since p is a fixed ~~pt~~ of "g" i.e. $g(p) = p$

$$g(p) = p - f(p) = p \Rightarrow f(p) = 0$$

(ii) $g(x) = x - c f(x)$

Similarly, if a function "g" has a fixed point "p" Then the function "f" defined by $f(x) = x - g(x)$ has a zero at "p"

Remarks:

- Fixed point form of root finding is easier to analyze.
- Certain choice lead to very powerful root finding techniques.

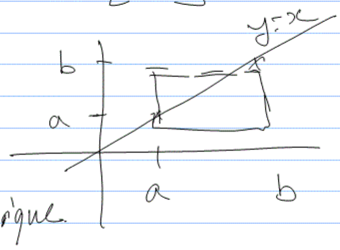
Existance and uniquness of a fixed point

Thorem:

(i) If $g \in C[a,b]$ and $g(x) \in [a,b]$ for all $x \in [a,b]$

Then g has a fixed point in $[a,b]$

(ii) If in addition, $g'(x)$ exists on (a,b)
with $|g'(x)| < 1$ $\forall x \in (a,b)$
Then the fixed point in $[a,b]$ is unique.



Remark:

The hypotheses of the above Theorem are sufficient to guarantee a unique fixed point but not necessary.

Ex: $$g(x) = \frac{x^2-1}{3} \quad \text{on } [3,4]$$

$$g(3) \notin [3,4]$$
$$g(4) \notin [3,4]$$
$$g'(4) = 8/3 > 1$$

but it has a unique fixed point $p = \frac{1}{2}(3+\sqrt{3}) = 3.3$

How? Suppose p is a fixed point

$$\Rightarrow \quad p = \frac{p^2-1}{3} \quad \Rightarrow p^2 - 3p - 1 = 0$$

Ex: Find the root of $f(x) = x^3 + 4x^2 - 10$ in $[1,2]$ using fixed point iteration.

- The function has unique root in $[1,2]$
- use $p_0 = 1.5$

$$f(x) = x^3 + 4x^2 - 10$$

(i) $x = g_1(x) = x - f(x) = x - x^3 - 4x^2 + 10$

(ii) $x = g_2(x) = \left(\frac{10}{x} - 4\right)^{1/2}$

(iii) $x = g_3(x) = \frac{1}{2}\left(10 - x^3\right)^{1/2}$

(iv) $x = g_4(x) = \left(\frac{10}{4+x}\right)^{1/2}$

(v) $x = g_5(x) = x - \frac{x^3 + 4x^2 - 10}{3x^2 + 8x}$

$x(x^2 + 4x) = 10$

$x^2 = \frac{10}{x} - 4x$

$x = \left(\frac{10}{x} - 4x\right)^{1/2}$

n	(i)	(ii)	(iii)	- -	v
0	1.5				
1	-0.875				

2	6.732
3	
4	
5	

(i) $x = g_1(x) = x - x^3 - 4x^2 + 10$

Let $p = g_1(p) = p - p^3 - 4p^2 + 10$

Let

$$p_n = p_{n-1} - p_{n-1}^3 - 4p_{n-1}^2 + 10, \quad p_0 = 1.5$$

Let $p_0 = 1.5$

$$p_1 = p_0 - p_0^3 - 4p_0^2 + 10$$

$$= -0.875 \ldots$$

$$p_2 = p_1 - p_1^3 - 4p_1^2 + 10$$

$$= 6.732$$

(i) — diverges

(ii) — involves root of a negative number

(iii) - Converges but slow

(iv) — ok,

(v) - good

Q: How to find a fixedpoint problem that produce a sequence that is reliable and converges rapidly to a solution for a given root finding problem.

Ans: Let $g \in C[a,b]$ be such that $g(x) \in [a,b]$ for all $x \in [a,b]$. Suppose in addition, $g'(x)$ exists in (a,b) and a constant $0 < K < 1$ exists with

$$|g'(x)| \leq K \quad \forall\ x \in (a,b)$$

Then, for any P_0 in $[a,b]$, the sequence defined by

$$P_n = g(P_{n-1}) \quad n \geq 1$$ converges to the unique fixed point p in $[a,b]$.

Consider (i) $g_1(x) = x - x^3 - 4x^2 + 10 \quad x \in [1,2]$

$$g_1(1) = 6; \quad g_1(2) = -12$$

$$\Rightarrow \quad g_1(x) \notin [1,2]$$

ie g_1 does not map $[1,2]$ into itself.

$$g_1'(x) = 1 - 3x^2 - 8x$$

$$|g_1'(x)| > 1 \quad \forall\ x \in (a,b)$$

Although, the hypothesis of the theorem does not guarantee that

the method must fail when the hypothesis is not satisfied
But there is no reason to expect that "g_1" Converges.

Case (ii) $\quad g_2(x) = \left(\frac{10}{x} - 4x\right)^{1/2}$

$g_2(x) \notin [1,2] \quad \forall\, x \in [1,2]$

$g_2'(x) = \left(-\frac{10}{x^2} - 4\right) \Big/ \left(2\sqrt{\frac{10}{x} - 4x}\right)$

$|g_2'(x)| > 1 \quad \forall\, x \in (1,2)$

$\not< 1$

Case (iii) $\quad g_3(x) = \frac{1}{2}(10 - x^3)^{1/2}$

$g_3(1) = 1.5, \quad g_3(2) = 0.707$

$g_3'(x) = -\frac{3}{4}x^2(10-x^3)^{-1/2}$

$|g_3'(x)| \not< 1 \quad \forall\, x \in (1,2)$

$g_3'(2) \simeq 2.12$

$|g_3'(1)| = 1/4 < 1$

$$\left| g_3^{'}(1.5) \right| = 0.65 < 1$$

Let us consider $[1, 1.5]$ instead of $[1, 2]$

$\Rightarrow$ $g_3(x)$ satisfies the hypothesis of the Theorem.

Ex: Case (iv) & (v)

Newtons method (Newton-Raphson method)

Suppose $f \in C^2[a,b]$. Let $P_0 \in [a,b]$ be an approximation of p of $f(x) = 0$ such that $f'(P_0) \neq 0$ and $|P - P_0|$ is small

Consider the first Taylor polynomial for $f(x)$ about P_0 with $x = p$.

$$f(P) = f(P_0) + (P - P_0) f'(P_0) + \frac{(P - P_0)^2}{2!} f''(\xi(P_0))$$

where ξ lie between p and p_0.

Since $f(p) = 0 \Rightarrow$

$$0 = f(p_0) + (p - p_0) f'(p_0) + \frac{(p-p_0)^2}{2!} f''(\xi)$$

Since $|p - p_0|$ is small $\Rightarrow (p-p_0)^2$ term will also be smaller.

$$0 = f(p_0) + (p - p_0) f'(p_0)$$

$$\Rightarrow \quad p - p_0 = -\frac{f(p_0)}{f'(p_0)}$$

$$\doteq$$

$$\Rightarrow \quad p = p_1 = p_0 - \frac{f(p_0)}{f'(p_0)}$$

which gives the sequence

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad n = 1, 2, ----$$

Remarks:

- Newtons method cannot be continued when $f'(p_{n-1}) = 0$

- If P_0 is not sufficiently close to the actual root, the Newton's method may not Converge.

Ex: (i) $f(x) = x^2 - 6$. and $P_0 = 1$ find P_3 using Newton.

$f'(x) = 2x$

$$P_1 = P_0 - \frac{f(P_0)}{f'(P_0)} = 3.5$$

$P_2 = 2.607$

$P_3 = 2.4542$

(ii) $f(x) = -x^3 - \cos x$ with $P_0 = -1$

$P_1 = -0.8803$

$P_3 = -0.865$

Try with $P_0 = 0$!!

(iii) $f(x) = x^2 - 10 \cos x$ with

(a) $P_0 = -100$ & (b) $P_0 = 100$

n	(a) p_n	(b) p_n
1	-48.74	
2	-21.5967	
3	-11.48	
4	-2.48	
5	-1.2	
⋮	⋮	
	-1.3793	1.3793

Relative error:

$$\frac{|p_n - p_{n-1}|}{|p_{n-1}|} < \varepsilon$$

Secant method.

– Major weakness of Newton's method is that it needs the value of the derivative of the function at each approximation.

By definition:

$$f'(p_{n-1}) = \lim_{x \to p_{n-1}} \frac{f(x) - f(p_{n-1})}{x - p_{n-1}}$$

Let $x = P_{n-2}$

$$f'(P_{n-1}) \simeq \frac{f(P_{n-2}) - f(P_{n-1})}{P_{n-2} - P_{n-1}} = \frac{f(P_{n-1}) - f(P_{n-2})}{P_{n-1} - P_{n-2}}$$

Using this approximation for $f'(P_{n-1})$ in the Newton's method it gives

$$P_n = P_{n-1} - \frac{f(P_{n-1})\,(P_{n-1} - P_{n-2})}{f(P_{n-1}) - f(P_{n-2})}$$

This technique is called the "Secant method".

Remark:

- it needs two initial values.
- Root braketting is not gurantted for either Newton or Secant methods.
- If P_0 is not given, use a few iterations of the Bisection method and use approximation as a initial guess for Newtons

Ex: 1 $f(x) = \cos x - x$ with $P_0 = 0.5$, $P_1 = \pi/4$.

$P_0 = \pi/4$

n	Newton	Secant
0	0.7853	0.5
⋮		
5	0.73908	0.73908

Error estimate:

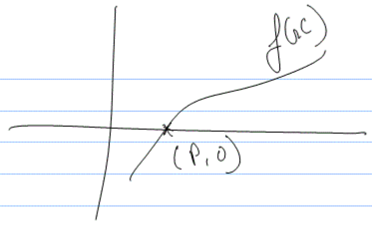
Assume that the fixed point condition hold ie for $p \in [a,b]$

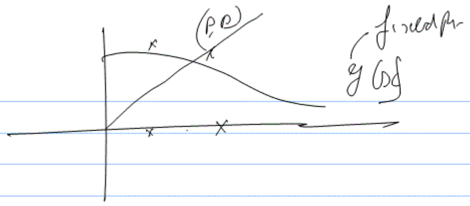
(1) $g(x) \in C[a,b]$

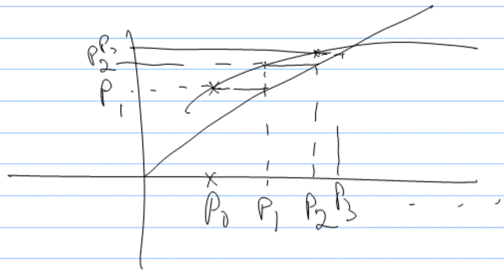
(2) $g(x) \in [a,b]$

(3) $|g'(x)| < 1 \quad \forall\, x \in (a,b)$

Root finding $\Rightarrow$ $f(p) = 0$







Error:

$$P_n = g(P_{n-1}) \quad \text{(approximation)}$$

$$P = g(P)$$

$$|P - P_n| = |g(P) - g(P_{n-1})|$$

$$= |g'(c)|\,|P - P_{n-1}|$$

MVT

$$f'(c) = \frac{f(b)-f(a)}{(b-a)}$$

$c \in [a, b]$

$$\varepsilon_n = |g'(c)|\,\varepsilon_{n-1}$$

where c lie inbetween P and P_{n-1}

Since $|g'(x)| \leq K < 1$ holds for $\forall x\ [a,b]$

$$\Rightarrow \quad \varepsilon_n = K\,\varepsilon_{n-1}$$

$$\vdots$$

$$\Rightarrow \quad \varepsilon_n = K^n\,\varepsilon_0$$

$$\lim_{n\to\infty} \varepsilon_n \to 0 \qquad (\because K < 1)$$

Remark:

(i) The smaller $|g'(x)| \Rightarrow$ faster Convergence

(ii) divergence occurs for $|g'(x)| > 1 \quad \forall x \in [a,b]$

(iii) $|g'(x)| < 1$ is sufficient but not necessary

Order of Convergence:

Suppose $\lim_{n\to\infty} P_n \to P$ with $P_n \neq P$ for all n.

If there exist positive Constants λ & K with

$$\lim_{n\to\infty} \frac{|P_{n+1} - P|}{|P_n - P|^{\alpha}} = \lim_{n\to\infty} \frac{|\varepsilon_{n+1}|}{|\varepsilon_n|^{\alpha}} = \lambda,$$

Then $\{P_n\}_{n=0}^{\infty}$ Converges to "p" with the order α and

The asymptotic error constant λ.

Eg: $\alpha = 1$ ie $\lim\limits_{n \to \infty} \frac{|\varepsilon_{n+1}|}{|\varepsilon_n|} = \lambda \Rightarrow$ linear

$\alpha = 2$ Quadratic

$\alpha = 3$ Cubic

Order of Convergence for fixed point iteration

$$P = g(P) \quad \text{(exact)}$$
$$P_{n+1} = g(P_n) \quad \text{(approximation)}$$

$$P_{n+1} - P = \underline{\underline{g(P_n)}} - g(P) \quad —(1)$$

Now expand $g(P_n)$ using Taylor's series about "p"

$$g(P_n) = g(P) + (P_n - P)\, g'(P) + \frac{(P_n - P)^2}{2!} g''(P) + \cdots$$

Using it in (1), we get

$$P_{n+1} - P = g'(P)(P_n - P) + \frac{g''(P)}{2!}(P_n - P)^2 + \frac{g'''(P)}{3!}(P_n - P)^3 + \cdots$$

$$\underline{\varepsilon_{n+1} = g'(P)\, \varepsilon_n} + \frac{g''(P)}{2!}\varepsilon_n^2 + \cdots$$

(i) $\lim_{n\to\infty} \varepsilon_{n+1} = K\varepsilon_n$, where $K < 1$

$\Rightarrow$ linear

(ii) Suppose $g'(P) = 0$ $\Rightarrow$

$$\lim_{n\to\infty} \varepsilon_{n+1} = K\varepsilon_n^2, \quad \text{where } K < 1$$

$\Rightarrow$ Quadratic Convergence.

Thus, in general, we need $g^m(x) = 0$ for all $m = 1, 2, \ldots n$ to get $\lim_{n\to\infty} \varepsilon_{n+1} = \frac{g^{n+1}}{(n+1)!} \varepsilon_n^{n+1}$

$\Rightarrow$ $(n+1)$ Order of Convergence!

Convergence of Newton's method:

If $\underline{g(x) = x - \phi(x) f(x)}$

Scaling function — $\phi(x)$; function whose Root(s) need to be Calculated — $f(x)$

Then $g(P) = P - \phi(x) \underbrace{f(P)}_{0}$

$= P$

$\Rightarrow$ $g(x)$ is a function with a "fixed point" at the root of "

$f(x)$. ie $g(p) = p$ and $f(p) = 0$.

Now, to construct quadratic convergence scheme, we need $\underline{g'(p) = 0}$.

$$g'(x) = 1 - \left[\phi'(x) f(x) + f'(x) \phi(x) \right]$$

$$g'(p) = 1 - \left[\phi'(p) \underbrace{f(p)}_{0} + f'(p) \phi(p) \right]$$

$$g'(p) = 1 - f'(p) \phi(p)$$

But we need $g'(p) = 0$

$$\Rightarrow \quad g'(p) = 1 - f'(p) \phi(p) = 0$$

ie $\phi(p) = \dfrac{1}{f'(p)}$ (assuming $f'(p) \neq 0$

Recall

$$g(x) = x - \phi(x) f(x)$$

$$= x - \frac{f(x)}{f'(x)}$$

Thus, in general the fixed point iteration becomes

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)} \Rightarrow \text{Newton's method.}$$

Remark: Guaranteed Quadratic Convergence for some P_0 Provided $f'(P) \neq 0$.

What if $f'(P) = 0$ and $f''(P) \neq 0$ in Newtons method.

$$\text{Recall } \quad g(x) = x - \frac{f(x)}{f'(x)}$$

We derived it by assuming $g'(P) = 0$ (Quadratic Convergence)

$$\text{Now} \quad g'(x) = 1 - \left[\frac{f'f' - ff''}{[f']^2} \right]$$

$$g'(x) = \frac{ff''}{[f']^2}$$

Check $g'(P)$?

$$g'(P) = \frac{f(P)\, f''(P)}{[f'(P)]^2} = \frac{0}{0} \quad \text{undefined.}$$

However, by L'Hopital rule:

$$\lim_{x \to p} \frac{f''(x) f(x)}{[f'(x)]^2} = \lim_{x \to p} \frac{f'f'' + \cancel{ff'''}^{\,0}}{2f'f''}$$

$$\lim_{x \to p} \frac{f}{g} = \lim_{x \to p} \frac{f'}{g'}$$

$$= \lim_{x \to p} \frac{f'(x) f''(x)}{2 f'(x) f''(x)} = \frac{1}{2}$$

$$\Rightarrow g'(p) \longrightarrow \frac{1}{2} \neq 0$$

$\Rightarrow$ we get only linear Convergence.

Also, we should be carefull in Computations using Computers!

Zero of Multiplicity

A Solution "p" of $f(x) = 0$ is a Zero of multiplicity "m" of f if for $x \neq p$, we can write $f(x) = (x-p)^m q(x)$, where $\lim_{x \to p} q(x) \neq 0$.

Remark:

$q(x)$ represents the portion of $f(x)$ that does not Contribute to the zero of "f".

One method of handling the problem of multiple roots is to define

$$\mu(x) = \frac{f(x)}{f'(x)}$$

If P is a zero of f of multiplicity "m" and

$$f(x) = (x-P)^m q(x), \text{ then}$$

$$\mu(x) = \frac{(x-P)^m q(x)}{m(x-P)^{m-1} q(x) + (x-P)^m q'(x)}$$

$$= (x-P) \frac{q(x)}{m\,q(x) + (x-P)\,q'(x)}$$

It is also having zero at P. However

$$\frac{q(P)}{m\,q(P) + 0} = \frac{\cdot}{m} = \frac{1}{m} \neq 0$$

& P is simple zero of $\mu(x)$

Now, Newton's method can be applied to $\mu(x)$ to get

$$g(x) = x - \frac{\mu(x)}{\mu'(x)} = x - \frac{f/f'}{[f']^2 - ff''] / [f']^2}$$

$$g(x) = x - \frac{f f'}{[f']^2 - f f''}$$

which is the modified Newton's method.

Ex:

Solve $f(x) = e^x - x - 1$, with $P_0 = 1$ using both Newton's and modified Newtons method.!

Remark: If the numerator terms $f'(x) \simeq 0$ and $f(x) f''(x) \simeq 0$, it lead to instability in Computer due to roundoff error.

Accelerated Convergence:

Aitken's method

Motivation:

Construct a new sequence $\{\tilde{P}_n\}_{n=0}^{\infty}$ that Converges more rapidly to "p" than does $\{p_n\}_{n=0}^{\infty}$.

Assume that the signs of $p_n - p$, $p_{n+1} - p$ and $p_{n+2} - p$ agree.

we have $\lim_{n\to\infty} \boxed{\varepsilon_{n+1} = g'(p)\,\varepsilon_n}$ (linear)

$$\lim_{n\to\infty} (p_{n+1} - p) = g'(p)(p_n - p) \quad \text{—} \ (*)$$

$$\text{Illy} \quad \lim_{n\to\infty} (p_{n+2} - p) = g'(p)(p_{n+1} - p)$$

$$\Rightarrow \quad p_{n+2} - p_{n+1} = g'(p)\,(p_{n+1} - p_n)$$

$$\Rightarrow \quad g'(p) \simeq \frac{p_{n+2} - p_{n+1}}{p_{n+1} - p_n} \qquad (\text{assumption } n \gg 1)$$

use it in $(*)$ to get

$$p_{n+1} - p = \frac{p_{n+2} - p_{n+1}}{(p_{n+1} - p_n)}(p_n - p)$$

Solve for "p"

$$p_{n+1}^2 - (p_{n+1} - p_n)p - p_{n+1}p_n = (p_{n+2} - p_{n+1})p_n - (p_{n+2} - p_{n+1})p$$

$$(p_{n+2} - 2p_{n+1} + p_n)\,p = (p_{n+2} - \cancel{p_{n+1}})p_n + \cancel{p_{n+1}p_n} - p_{n+1}^2$$

$$= p_{n+2}p_n - p_{n+1}^2$$

$$= \left(P_{n+2} P_n - 2P_{n+1} P_n + P_n^2 \right) + 2P_{n+1} P_n - P_n^2 - P_{n+1}^2$$

$$= \left(P_{n+2} - 2P_{n+1} + P_n \right) P_n - \left(P_{n+1} - P_n \right)^2$$

$$\boxed{\tilde{P}_n = P_n - \frac{(P_{n+1} - P_n)^2}{(P_{n+2} - 2P_{n+1} + P_n)}}$$

P_0

$P_1 \longrightarrow \tilde{P}_0$

$P_2 \longrightarrow \tilde{P}_1$

P_3

P_4

Exc:

Let $P_n = \cos(1/n)$, w.r.t $P_n \xrightarrow{\text{linearly}} P$

n	P_n	$\tilde{P}_n$
1		
2		
$\vdots$		

Steffensen's method

$+^{ve}$ - use $\tilde{P}_0$ to compute $P_3 := g(\tilde{P}_0)$, assuming that $\tilde{P}_0$ is a better guess than P_2

- method converges quadratically

$-^{ve}$ * needs more calculations.

$$y = g(x) = x^2 + 2x + 3$$

Multivariable nonlinear equations:

We can generalize the fixed point concepts that we developed for single variable nonlinear equations to the system of multivariable equations.

Consider the system of equations:

$$f_1(x_1, x_2, \ldots, x_N) = 0$$
$$f_2(x_1, x_2, \ldots, x_N) = 0$$
$$\vdots$$

$$f_N(x_1, x_2, \dots, x_N) = 0$$

Denote it as

$$\bar{F}(\bar{x}) = 0$$

The solution is $\bar{p} = (p_1, p_2, \dots p_N)$

Let $\bar{p} = (p_1, p_2, \dots p_N)$ be a fixed point vector of $\bar{G}(\bar{p})$, ie

$$\bar{G}(\bar{p}) = \bar{p}$$

It is true $\Longleftrightarrow$ $g_i(p_1, p_2, \dots, p_N) = p_i \;\; \forall \; i = 1, 2, \dots N$

where $\bar{G} = (g_1, g_2, \dots, g_N)$

The existance of $\bar{p}$ is guaranteed by

(i) $\bar{G}(\bar{x})$ is continuous on $a_i \le x_i \le b_i$, $i = 1, \dots, N$

(ii) $\bar{G}(x)$ is bounded on domain

$$a_i \le g_i(\bar{x}) \le b_i$$

for $a_i \le x_i \le b_i$, $i = 1, \dots, N$

(iii) $\bar{p}$ is unique if in addition to (i) & (ii)

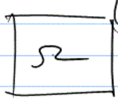
$$\sum_{j=1}^{N} \left| \frac{\partial g_i(\bar{x})}{\partial x_j} \right| < 1 \text{ for } i = 1, 2, \dots, N$$

Eg $N = 2$

$a_1 = 0 = a_2$

$b_1 = 1 = b_2$

(1,1)



(0,0)

$\bar{p} = (p_1(x,y), p_2(x,y))$

Convergence Rate:

linear if $\dfrac{\partial g_i(\bar{p})}{\partial x_j} \neq 0 \quad \forall\ i,j = 1, 2, \ldots, N$

Quadratic if $\dfrac{\partial g_i(\bar{p})}{\partial x_j} = 0$

Newton's method for system of Equations

Recall $g(x) = x - \phi(x) f(x)$

For Newton's method, it requires $\phi(x) = \dfrac{1}{f'(x)}$

$$\Rightarrow g(x) = x - \frac{f(x)}{f'(x)}$$

For multivariable

$$\bar{G}(\bar{x}) = \bar{X} - \frac{F(\bar{x})}{F'(\bar{x})}$$

where $$F'(\bar{x}) = D\bar{F} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & & \frac{\partial f_2}{\partial x_N} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_N}{\partial x_1} & \frac{\partial f_N}{\partial x_2} & & \frac{\partial f_N}{\partial x_N} \end{bmatrix}$$

$$= J(\bar{x})$$

$$\Rightarrow \boxed{\bar{G}(\bar{x}) = \bar{x} - J^{-1}\bar{F}(\bar{x})}$$

$$\Rightarrow \begin{pmatrix} x_1^{k+1} \\ x_2^{k+1} \\ \vdots \\ x_N^{k+1} \end{pmatrix} = \begin{pmatrix} x_1^{k} \\ x_2^{k} \\ \vdots \\ x_N^{k} \end{pmatrix} - \underbrace{\overbrace{\begin{bmatrix} & & \\ & & \end{bmatrix}_{N\times N}}^{J^{-1}} \begin{pmatrix} f_1(\bar{x}^k) \\ f_2(\bar{x}^k) \\ \vdots \\ f_N(\bar{x}^k) \end{pmatrix}_{N\times 1}}_{\bar{y}}$$

Let $$\bar{y} = J^{-1}\bar{F}(\bar{x})$$

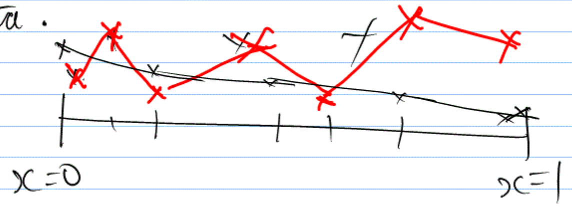
$$\Rightarrow J\bar{y} = \bar{F}(\bar{x}) \Longleftarrow$$

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial x_1} & \cdots & \frac{\partial f_N}{\partial x_N} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} f_1(\bar{x}) \\ \vdots \\ f_N(\bar{x}) \end{bmatrix}$$

- Solve $N \times N$ system of algebraic equations to get $\bar{Y}$
- Then update the iteration: $\bar{X}^{k+1} = \bar{X}^{k} - \bar{Y}^{k}$

Interpolation:

The process of obtaining a function that fits the given data.



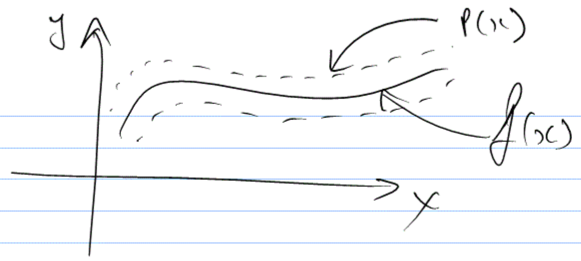
Polynomial Interpolation

f^n - easy to evaluate
- easy to integrate/differentiate.

The approximation is based on the Weierstrass principle.

Weierstrass approximation Theorem:
Suppose that f is defined and continuous on $[a,b]$. For each $\varepsilon > 0$ $\exists$ a polynomial $P(x)$ with the property that

$$|f(x) - P(x)| < \varepsilon \quad \forall x \in [a,b]$$



However, it does not tell us "how to find $P(x)$".

(I) one type of polynomial is a Taylor polynomial:

(i) Good for analysis since the truncation error is explicit, ie $\dfrac{f^{n+1}(\xi)\,(x-x_0)^{n+1}}{(n+1)!}$, $\xi \in (x, x_0)$

(ii) All information about $f(x)$ is concentrated at x_0
(all derivatives evaluated at "x_0")

(iii) Locally very good, but gets worse as $|x-x_0|$ grows

(4) A better polynomial (for interpolation) $P_n(x)$ matches $f(x)$ at a finite number of values.

(i) n^{th} order polynomial has $(n+1)$ coefficents.

(ii) $P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$

(iii) If $f(x_i) = P_n(x_i)$, then we need $n+1$ x_i's to find a unique set of coeff. $\{a_i\}$

Eg: Given x_1, x_2, x_3 and $f(x_1), f(x_2), f(x_3)$

Find a_0, a_1 and a_2 such that $P_2(x) = a_2 x^2 + a_1 x + a_0$ satisfies

$$P_2(x_i) = f(x_i) \quad \forall\ i = 1, 2, 3.$$

$$P_2(x_1) = a_2 x_1^2 + a_1 x_1 + a_0 = f(x_1) = f_1$$

$$P_2(x_2) = a_2 x_2^2 + a_1 x_2 + a_0 = f(x_2) = f_2$$

$$P_2(x_3) = a_2 x_3^2 + a_1 x_3 + a_0 = f(x_3) = f_2$$

$$\begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

$$Ax = B$$

$$\Rightarrow \quad x = A^{-1}B$$

Remarks:

(i) Given $(n+1)$ distinct samples of $f(x)$ ie $f(x_i)$, $i=0,\dots n$. we can always find a unique polynomial $P_n(x)$ of order "n" $\ni$ $P_n(x_i) = f(x_i)$, $i=0,\dots,n$.

(ii) We can show that the sytem generated for "$n+1$" Coeff is linear.

(iii) It is tedious: we want a quick way to get $\{a_i\}$ – the set of Coefficients.

(iv) The idea is to use a basis polynomial, ie define

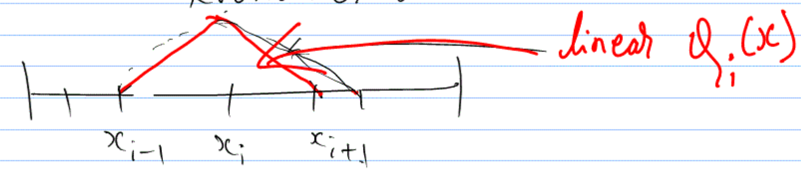
$$P_n(x) = \sum_{j=0}^{n} \underline{p_j}\, \underset{\uparrow \text{basis}}{Q_j(x)}$$

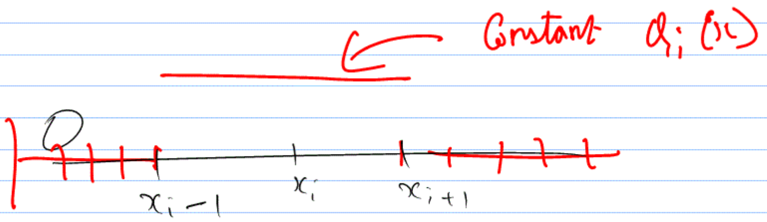
Note if $p_j = a_j$ and $Q_j(x) = x^j$, $j=0,\dots n$

$\Rightarrow$ standard polynomial of order "n".

(v) By cleverly choosing $\{Q_j(x)\}$ we can identify $\{p_j\}$ easily.

Let $Q_j(x_i) = \underset{\substack{\uparrow \\ \text{Kronocker delta}}}{\delta_{ij}} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases}$





Then $\quad \underline{P_n(x_i)} = \sum_{j=0}^{n} P_j\, q_j(x_i)$

$$= \sum_{j=0}^{n} P_j\, \delta_{ij} = P_i = \underline{f(x_i)}.$$

It leads to a number of possible basis functions with the property that

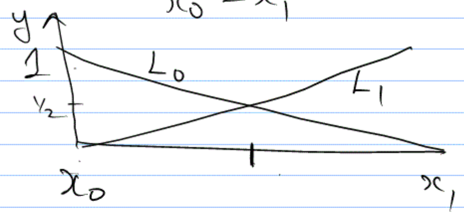
$$q_j(x_i) = \delta_{ij}$$

Note: q_j is having only local support.

one example is the <u>Lagrange Polynomials</u>

Determine a polynomial of degree one that passes through the distinct points (x_0, y_0) and (x_1, y_1) which is equivalent to approximating a function "f" for which $f(x_0) = y_0$ and $f(x_1) = y_1$, by a first-degree polynomial interpolating with the values of f.

Define $L_0(x) = \dfrac{x - x_1}{x_0 - x_1}$ & $L_1(x) = \dfrac{x - x_0}{x_1 - x_0}$;



then define

$$P(x) = L_0(x) f(x_0) + L_1(x) f(x_1) \quad \longleftarrow \; 1$$

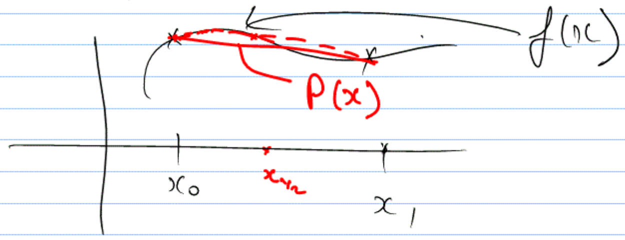
Since $L_0(x_0) = 1 \qquad L_1(x_0) = 0$

$L_0(x_1) = 0 \qquad L_1(x_1) = 1$

we have $P(x_0) = 1 \cdot f(x_0) + 0 \cdot f(x_1) = f(x_0) = y_0$

$P(x_1) = 0 \cdot f(x_0) + 1 \cdot f(x_1) = f(x_1) = y_1$

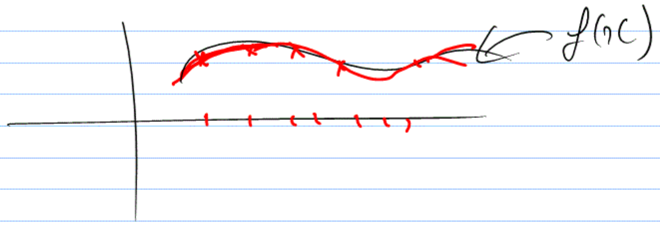
So P is the unique linear function passing through (x_0, y_0) and (x_1, y_1).



Generalize the concept of linear interpolation to construct higher order interpolation.

Consider the construction of polynomial of degree at most "n" that passes through "n+1" points

$(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$



In this case, we have to construct a function $L_{n,k}(x)$ with the property that

index for point, point

$$L_{n,k}(x_i) \begin{cases} = 0 & \text{if } i \neq k \\ \neq 0 & \text{if } i = k \end{cases}$$

$$\Rightarrow \quad L_{n,k}(x) = \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{(x - x_i)}{(x_k - x_i)}$$

$$= \frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdot(x_k-x_n)}$$

Thorem:

If $x_0, x_1, \ldots x_n$ are "$n+1$" distinct numbers and f is a function whose values are given at these numbers, then a unique polynomial $P(x)$ of degree at most "n" exists with

$$f(x_k) = P(x_k), \quad k=0,\ldots,n$$

This polynomial is given by

$$P(x) = f(x_0)L_{n,0}(x) + f(x_1)L_{n,1}(x) + \cdots + f(x_n)L_{n,n}(x)$$
$$= \sum_{k=0}^{n} f(x_k)\, L_{n,k}(x)$$

Where

$$L_{n,k}(x) = \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{(x-x_i)}{(x_k-x_i)}$$

Note $L_{n,k}(x)$ simply written as $L_k(x)$!

Ex: Let $x_0 = 2$, $x_1 = 2.5$, & $x_2 = 4$. Find second interpolating polynomial for $f(x) = 1/x$.

$$P_2(x) = f(x_0)\,L_0(x) + f(x_1)\,L_1(x) + f(x_2)\,L_2(x)$$

$$f_0 = f(2) = 1/2 \; ; \; f_1 = f(2.5) = 1/2.5 \; ; \; f_3 = 1/4$$

$$L_0(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} = (x - 6.5)x + 10$$

$$L_1(x) = \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} = \frac{(-4x+24)x - 32}{3}$$

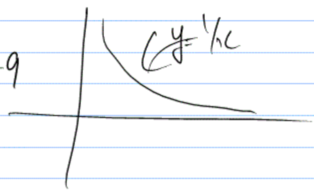
$$L_2(x) = \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} = \frac{(x-4.5)x + 5}{3}$$

Lagrange

$$\tilde{P}(x) = 1.116x^2 - 6.8215x + 9.679$$

$$\hat{P}(x) = 0.05x^2 - 0.425x + 1.15$$

direct way



Error :

Theorem: Suppose $x_0, x_1, \ldots, x_n$ are distinct numbers in the interval $[a,b]$ and $f \in C^{n+1}[a,b]$. The for each x in $[a,b]$, a number $\xi(x)$ [generally unknown] in (a,b) exists with

$$f(x) = P(x) + \frac{f^{n+1}(\xi(x))}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n) \quad (1)$$

Here $P(x)$ is the polynomial obtained from Lagrangian interpolation

Proof: Note that if $x = x_i$ then

$$f(x_i) = P(x_i) + \frac{f^{n+1}(\xi(x))}{(n+1)!}(x_i - x_0)\cdots\underbrace{\cancel{(x_i - x_i)}}_{0}\cdots(x_i - x_n)$$

$$\Rightarrow f(x_i) - P(x_i) = 0$$

and chosing $\xi(x_k)$ arbitrarly in (a,b) yields (1)

If $x \neq x_i$, then define a function $g(t)$ for $t \in [a,b]$ by

$$g(t) = f(t) - P(t) - [f(x) - P(x)] \prod_{i=0}^{n} \frac{(t-x_i)}{(x-x_i)} \quad (2)$$

Since $f \in C^{n+1}[a,b]$ and $P \in C^{\infty}[a,b]$

$\Rightarrow \quad g \in C^{n+1}[a,b]$.

Also for $t = x_k$, we have

$$g(x_k) = f(x_k) - P(x_k) - \left[f(x) - P(x)\right] \prod_{i=0}^{n} \frac{\overset{0}{\cancel{(x_k - x_i)}}}{\cancel{(x - x_i)}}$$

$$= 0$$

Further, for $t = x$

$$g(x) = f(x) - P(x) - \left[f(x) - P(x)\right] \overset{1}{\cancel{\prod_{i=0}^{n} \frac{(x - x_i)}{(x - x_i)}}}$$

$$= 0$$

Thus $g \in C^{n+1}[a,b]$ and "g" is zero at "$n+2$" points, $x, x_0, x_1, \ldots x_n$. By Generalized Rolle's Theorem:

> Supp $f \in C[a,b]$ is n times differentiable on (a,b). If $f(x)$ is zero at the "$n+1$" distinct numbers, $x_0, \ldots x_n$ in $[a,b]$, then a number $c \in (a,b)$ exists with
> $f^{n}(c) = 0$

There exists a number $\xi \in (a,b)$ for which $g^{n+1}(\xi) = 0$.

$$\Rightarrow \quad 0 = g^{n+1}(\xi) = f^{n+1}(\xi) - P^{n+1}(\xi) - \left[f(x) - P(x)\right] \frac{d^{n+1}}{dt^{n+1}} \left[\prod_{i=0}^{n} \frac{(t-x_i)}{(x-x_i)}\right]$$

Since $P(x)$ is a polynomial of order at most "n", the $(n+1)^{th}$ derivative, $P^{n+1}(x)$ is identically zero.

Also $\prod_{i=0}^{n} \frac{(t-x_i)}{(x-x_i)}$ is a polynomial of degree "$n+1$",

$$\Rightarrow \prod_{i=0}^{n} \frac{(t-x_i)}{(x-x_i)} = \left[\frac{1}{\prod_{i=0}^{n}(x-x_i)}\right] t^{n+1} + \left[\text{lower-degree terms in } t\right]$$

and $$\frac{d^{n+1}}{dt^{n+1}}\left(\prod_{i=0}^{n} \frac{(t-x_i)}{(x-x_i)}\right) = \frac{(n+1)!}{\prod_{i=0}^{n}(x-x_i)} + 0$$

Substituting these derivations, we get

$$0 = f^{n+1}(\xi) - \left[f(x) - P(x)\right] \frac{(n+1)!}{\prod_{i=0}^{n}(x-x_i)}$$

Solving for $f(x)$, we get

$$f(x) = P(x) - \frac{f^{n+1}(\xi(x))}{(n+1)!} \prod_{i=0}^{n}(x-x_i)$$

where $\xi \in (a,b)$.

Remark:

- Note that the error form for the Lagrange polynomial is quite similar to that for the Taylor polynomial.

- The n^{th} Taylor polynomial about x_0 has the error term of the form

$$\frac{f^{n+1}(\xi(x))}{(n+1)!}(x-x_0)^{n+1}$$

- The Lagrange polynomial of degree "n" uses the information at "n+1" distinct points, $x_0, x_1, x_2, \ldots, x_n$, in place of $(x-x_0)^{n+1}$,

ie
$$\frac{f^{n+1}(\xi)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n)$$

Q: Does there exist a set of points $\{x_i\}$ for which the "optimal" interpolation is acheived for Lagrange polynomials.

- optimal is defined as minimizing the maximum error on $[a, b]$.

Ans: Use Chebyscher Polynomial

Error: $$f(x) - P(x) = \underbrace{\frac{f^{n+1}(\xi(x))}{(n+1)!}}_{\text{fixed, unknown } \xi \text{ (generally)}} \underbrace{\prod_{i=0}^{n} (x - x_i)}_{\text{minimize the maximum of this quantity.}}$$

$\Rightarrow$ Choose $\{x_i\}$ cleverly! How?

We can prove that a minimal maximum exists on $[-1, 1]$ if we use the roots of the $(n+1)^{th}$ order Chebyschev polynomials for $\{x_i\}$

Chebyschev Polynomial definition: (Trignometric form)

$$T_n(x) = \cos\left(n \cos^{-1}(x)\right) \qquad n \geq 0$$

Recurrence Relation:

$$T_0(x) = 1$$
$$T_1(x) = x$$

$$T_{n+1}(x) = 2x\,T_n(x) - T_{n-1}(x), \qquad n \geq 1$$

Now, how to choose $\{x_i\}$ in $[a,b]$

Use $x_i = \frac{a+b}{2} + \frac{b-a}{2}\,\bar{x}_i$, $i = 0, \ldots, n$

where $\bar{x}_i = \cos\left[\frac{(2i+1)\pi}{2(n+1)}\right]$

which is optimal sample pts for $f(x)$ on $[a,b]$.

Iterative interpolation:

– Obtain higher order polynomials from combinations of lower order polynomials.

Definition:

Let "f" be a function defined at $x_0, \ldots x_n$ and suppose that $m_1, \ldots, m_k$ are distinct integers with $0 \leq m_i \leq n$ for each "i". The Lagrange polynomial that agrees with $f(x)$ at "k" points $x_{m_1}, x_{m_2}, \ldots x_{m_k}$ is denoted by

$$P_{m_1, m_2, \ldots m_k}(x)$$

Eg:

$P_0(x) \Rightarrow \quad P(x_0) = f(x_0)$ ie $P(x_0)$ agrees with $f(x_0)$

$P_2(x) \Rightarrow \quad P(x_2) = f(x_2)$

$P_{0,1}(x) \Rightarrow \quad P(x_0) = f(x_0)$ & $P(x_1) = f(x_1)$

Let x_0 & x_1 be given points, define

$$\begin{aligned} P_{0,1}(x) &= f(x_0) P_0(x) + f(x_1) P_1(x) \\ &= f(x_0) \frac{(x - x_1)}{(x_0 - x_1)} + f(x_1) \frac{(x - x_0)}{(x_1 - x_0)} \\ &= \frac{f(x_1)(x - x_0) - f(x_0)(x - x_1)}{(x_1 - x_0)} \\ &= \frac{P(x_1)(x - x_0) - P(x_0)(x - x_1)}{(x_1 - x_0)} \end{aligned}$$

$$P_{0,1}(x) = \frac{P_1(x)(x-x_0) - P_0(x)(x-x_1)}{x_1 - x_0}$$

lly $$P_{1,2}(x) = \frac{P_2(x)(x-x_1) - P_1(x)(x-x_2)}{x_2 - x_1}$$

lly $$P_{0,1,2}(x) = f(x_0)P_0(x) + f(x_1)P_1(x) + f(x_2)P_2(x)$$

$\bar{\text{ie}}$ matches $f(x)$ at x_0, x_1, x_2

$$= f(x_0)\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f(x_1)\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}$$

$$+ f(x_2)\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}$$

$$\Rightarrow P_{0,1,2}(x) = \frac{\underline{P_{1,2}(x)}(x-x_0) - P_{0,1}(x)(x-x_2)}{(x_2-x_0)}$$ Exercise

x_0	P_0				
		$P_{0,1}$			
x_1	P_1		$P_{0,1,2}$		
		$P_{1,2}$		$P_{0,1,2,3}$	
x_2	P_2		$P_{1,2,3}$		$P_{0,1,2,3,4}$
		$P_{2,3}$		$P_{1,2,3,4}$	
x_3	P_3		$P_{2,3,4}$		
		$P_{3,4}$			
x_4	P_4				

$\Rightarrow$ Higher order Polynomial.

This approach is called Neville's method :

Divided Difference:

The divided difference of "f" with respect to $x_0, x_1, \ldots, x_n$ are used to express $P_n(x)$ as

$$P_n(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) + \cdots$$

$$\cdots + a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

for appropriate Constants a_i, $i = 0, \ldots n$

Evaluation of the Co-eff $\{a_i\}$

Evaluating $P_n(x)$ at x_0 leads to

$$P_n(x_0) = a_0 \quad \left(= f(x_0) = f_0\right)$$

Similary evaluating $P_n(x)$ at x_1 leads

$$P_n(x_1) = a_0 + a_1(x_1 - x_0)$$

$$= \underline{f_0 + a_1(x_1 - x_0) = f(x_1) = f_1}$$

$$\Rightarrow \quad a_1 = \frac{f_1 - f_0}{x_1 - x_0}$$

Introduce the divided difference notation

Zeroth divided difference : $f[x_i] = f(x_i)$

First divided difference : $f[x_i, x_{i+1}] = \dfrac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$

Second divided difference : $f[x_i, x_{i+1}, x_{i+2}] = \dfrac{f[x_{i+1}, x_{i+2}] - f[x_i, x_{i+1}]}{x_{i+2} - x_i}$

Note that:

$$P_n(x_0) = a_0 = f(x_0) = f[x_0]$$

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_0, x_1]$$

$$\vdots$$

$$a_k = f[x_0, x_1 \dots, x_k]$$

Hermite interpolation:

- Polynomials match "f" at $n+1$ point $\{x_i\}$, $i=0, \dots n$ in the interpotation defined in the previous cases

- we want to Construct polynomials that match "f" and "f'" at $(n+1)$ points $\{x_i\}$, $i=0,\dots,n$.

$\Rightarrow$ use Hermite interpolation

In the Hermite interpolation, we have $(2N+2)$ Conditions, ie $P_N(x_i) = f(x_i)$ and

$$P_N'(x_i) = f'(x_i) \quad \forall\ i=0,\dots,n$$

Define: the polynomial in basis form:

$$P_{2N+2}(x) = \sum_{j=0}^{N} f(x_j)\, H_{2N+1,j}(x) + \sum_{j=0}^{N} f'(x_j)\, \hat{H}_{2N+1,j}(x)$$

with the properties

$$H_{2N+1,j}(x_i) = \delta_{ij} \qquad \hat{H}_{2N+1,j}(x_i) = 0$$

$$H'_{2N+1,j}(x_i) = 0 \qquad \hat{H}'_{2N+1,j}(x_i) = \delta_{ij}$$

$$\forall\ i=0,\dots,n$$

$$\Rightarrow \quad P_{2N+2}(x_i) = \sum_{j=0}^{N} f(x_j)\,\delta_{ij} + 0$$
$$= f(x_i)$$
$$P'_{2N+2}(x_i) = \sum_{j=0}^{n} f'(x_j)\,\delta_{ij}$$
$$= f'(x_i)$$

Also, we can write, H & $\hat{H}$ interms of Lagrange polynomial.

$$H_{2N+1,j}(x) = \left[1 - 2(x-x_j)\,L'_{n,j}(x_j)\right] L^2_{n,j}(x)$$

$$\hat{H}_{2N+1,j}(x) = (x-x_j)\,L^2_{n,j}(x)$$

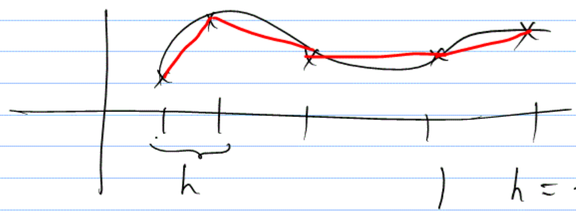
Error term : $\dfrac{f^{2N+2}(\xi(x))}{(2N+2)!} \prod_{j=0}^{N} (x-x_j)^2$

Remarks:

- Global polynomial interpolation:
- Single polynomial to approximate "$f(x)$", $x \in [a,b]$ over whole range
- advantages of having single polynomial are:
  * Storage, Computations, etc.
- wiggle can be a serious problem:
  * Chebyshev sampling may help.

- Another possibility is to use piecewise polynomial that is, polynomial over smallest segment of the interval $[a,b]$.
  * minimizes the potential unwanted wiggles

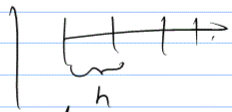
Eg: Piecewise-linear polynomial:



$h = \frac{b-a}{N}$ $\quad h < 1$

– It is simple but not a very accurate solution.

$$|P(x) - f(x)| \simeq O(h^2)$$

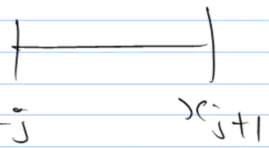


$\Rightarrow$ h must be small $\Rightarrow$ more sample points.

— Also $p'(x_i)$ is discontinuous for $\{x_i\}$, $i = 0, \ldots n$, which is not desirable in many cases.

So, one possibility is to use Hermit cubic polynomial for local interpolation (x_j, x_{j+1}) but it requires

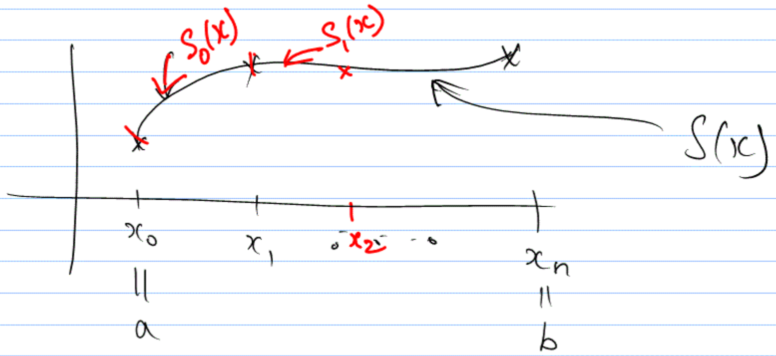
$$f(x_j),\ f'(x_j),\ f(x_{j+1}),\ f'(x_{j+1}).$$



— But the disadvantage is that the first derivatives are needed at end points.!

A useful compromise is to use a cubic spline interpolation:

Cubic spline interpolation:



Let $I=[a,b]$ be the given interval. Define

$$h_j = x_{j+1} - x_j$$

Given a function "f" on I, and a set of nodes $a = x_0 < x_1 < \cdots < x_n = b$.

A cubic spline interpolat "S" for the function "f" is a function that satisfies the following conditions.

(a) $S(x)$ is a cubic polynomial, denote $S_j(x)$ on the subinterval $[x_j, x_{j+1}]$ for $j=0, \ldots, n-1$.

(b) $S_j(x_j) = f(x_j)$ and $S_j(x_{j+1}) = f(x_{j+1})$

(c) $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$ $\quad j=0, \dots n-2$

(d) $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$ $\quad j=0, \dots n-2$

(e) $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$ $\quad j=0, \dots n-2$

(f) one of the following sets of boundary conditions is satisfied:

(i) $S''(x_0) = S''(x_n) = 0$ [free or natural BC]

(ii) $S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$ [clamped BC]

Construction of Cubic Spline:

Numerical Differintation

- looking for ways to approximate function derivatives with combination of function evaluations at discrete points
- Construct a polynomial approximation to $f(x)$ through evenly spaced samples of $f(x_i)$, then diffentiate the polynomial.
- Use the Taylor series expansion for $f(x)$ about x_j (where the derivative is desired)

ie $$f(x) = f(x_j) + f'(x_j)(x - x_j) + f''(x_j)\frac{(x-x_j)^2}{2!} + \cdots$$

Let $h = x_{j+1} - x_j$

$f_j = f(x_j)$

$x = x_{j+1}$

$$\Rightarrow f_{j+1} = f_j + h f_j' + \frac{h^2}{2} f_j'' + \cdots \qquad (1)$$

* If h is small (it should be) then the terms in expansion diminish.

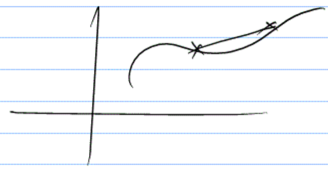
* The approximation gets worse as the distance from x_j increases.

From (1), solve for f'

$$f'_j = \frac{f_{j+1} - f_j}{h} + O(h)$$

$$\Rightarrow \text{error} \simeq \frac{h}{2} f''_j(\xi)$$ (note that it is not h^2, since we divide it by h)

This is approximation to first derivative with order "h" accuracy



Recall Secant Method:

$$f'(P_n) \simeq \frac{f(P_n) - f(P_{n-1})}{P_n - P_{n-1}}$$

Definition:

(r_1) $\quad \Delta f_j = f_{j+1} - f_j$ (First order forward difference)

$$f_j' = \frac{\Delta f_j}{h}.$$

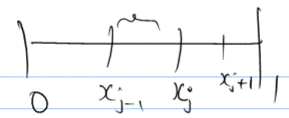
(ii) $\nabla f_j = f_j - f_{j-1}$ (First order Backward difference)

Use the same Taylor expansion about x_j but evaluate at x_{j-1}

$$\Rightarrow \quad f_j' = \frac{f_j - f_{j-1}}{h} + O(h)$$

where $h = x_j - x_{j-1}$

$$\boxed{f_j' = \frac{\nabla f_j}{h}.}$$



Second Derivative (Forward Difference)

Expand Taylor series about x_j and evaluate at x_{j+1}

$$f_{j+1} = f_j + h f_j' + \frac{h^2}{2!} f_j'' + \frac{h^3}{3!} f'''(\xi(x_{j+1})) \quad \text{——— (3)}$$

Again expand Taylor series about x_j and evaluate for $x = x_{j+2}$; Let $\boxed{2h = x_{j+2} - x_j}$

$$f_{j+2} = f_j + 2h f_j' + \frac{(2h)^2}{2!} f_j'' + \frac{(2h)^3}{3!} f'''(\xi(x_{j+2})) \quad \text{--- } ④$$

④ $- 2$ ③ $\Rightarrow$

$$f_{j+2} - 2f_{j+1} = -f_j + \underline{\underline{h^2 f_j''}} + ch^3$$

$$f_j'' = \frac{f_{j+2} - 2f_{j+1} + f_j}{h^2} + O(h)$$

Note:
$$\Delta^2 f_j = \Delta(\Delta f_j) = \Delta(f_{j+1} - f_j)$$
$$= (f_{j+2} - f_{j+1}) - (f_{j+1} - f_j)$$
$$= f_{j+2} - 2f_{j+1} + f_j$$

$$\Rightarrow \boxed{f_j'' = \frac{\Delta^2 f_j}{h^2} + O(h)}$$

Second derivative (Central Difference)

Expand Taylor series at "x_j" and evaluate for $x = x_{j+1}$; $h = x_{j+1} - x_j$

$$f_{j+1} = f_j + h f_j' + \frac{h^2}{2} f_j'' + \frac{h^3}{3!} f_j''' + \frac{h^4}{4!} f^{iv}(\xi(x_{j+1}))$$

Again at x_j and evaluate for $x = x_{j-1}$

$$f(x_{j-1}) = f(x_j) + f'(x_{j-1} - x_j) + \frac{f''(x_j)}{2!}(x_{j-1} - x_j)^2$$

$$+ \cdots$$

$$f_{j-1} = f_j - h f_j' + \frac{h^2}{2} f_j'' - \frac{h^3}{3!} f''' + \frac{h^4}{4!} f^{iv}(\xi)$$

Add the two expansions, to get

$$f_{j+1} + f_{j-1} = 2 f_j + h^2 f_j'' + \frac{h^4}{24}\left[f^{iv}(\xi_1) + f^{iv}(\xi_2)\right]$$

Solving for f'', we get

$$f_j'' = \frac{f_{j+1} - 2 f_j + f_{j-1}}{h^2} + O(h^2)$$

Notes:

— Forward difference: $f_j'' = \dfrac{\Delta^2 f_j}{h^2}$

— Backward difference: $\nabla^2 f_j = \nabla(\nabla f_j) = \nabla(f_j - f_{j-1})$

$$= (f_j - f_{j-1}) - (f_{j-1} - f_{j-2})$$

$$= f_j - 2f_{j-1} + f_{j-2}$$

$$\Rightarrow f_j'' = \frac{\nabla^2 f_j}{h^2}$$

— For f_j''' we need three Taylor expansions

Try it & expand Taylor series at x_j and evaluate for $x = x_{j+1}, x_{j+2}, x_{j+3}$

$$\Rightarrow f_j''' = \frac{f_{j+3} - 3f_{j+2} + 3f_{j+1} - f_j}{h^3} + O(h)$$

(or)

$$f_j''' = \frac{\Delta^3 f_j}{h^3}$$

ie Forward difference approximation of $f_j^{(1)}$ to $O(h)$

— In general we can construct approximation to n^{th} derivative of "f" at x_j (ie f_j^n) to accuracy $O(h)$ using weighted sums of "n" Taylor expansions with "$n+1$" function evaluations [Both Forward & Backward].

— In a similar way, we can construct higher order approximation to a derivative at the cost of more Taylor series expansion and more function evaluations.

— higher order of "h" $\Rightarrow$ higher accuracy

<u>Ex:</u>

(i) $f_j^{\prime}$ to $O(h^2)$

Expand Taylor series at "x_j" and evaluate for $x = x_{j+1}$ and $x = x_{j+2}$

$$f_{j+1} = f_j + h f_j' + \frac{h^2}{2} f_j'' + \frac{h^3}{3!} f'''(\xi_1) \quad \text{--- (1)}$$

$$f_{j+2} = f_j + 2h f_j' + \frac{(2h)^2}{2!} f_j'' + \frac{(2h)^3}{3!} f'''(\xi_2) \quad \text{--- (2)}$$

$4(1) - (2)$

$$4 f_{j+1} - f_{j+2} - 3 f_j = 2h f_j' - \frac{4}{3!} h^3 C$$

$$\Rightarrow \quad f_j' = \frac{-f_{j+2} + 4 f_{j+1} - 3 f_i}{2h} + O(h^2)$$

Remark:

- Higher order costs another function evaluation.
  $\Rightarrow$ we are fitting higher order polynomials to our function at x_i, so, we need more points.

- For forward and Backward n^{th} order derivative to accuracy $O(h^m)$ requires

$(n+m)$ points [function values] involving

$(n+m-1)$ Taylor series.

$$\|e\|_2 \leq c\, h^{k}$$

$k \leftarrow$ order.

$h \leftarrow$ mesh size.

$h \ll 1$

$k \geq 1$

Computer errors in numerical Differentiation.

Eg: Centeral difference of f_j' is of $O(h^2)$

Exact match:

$$f_j' = \frac{f_{j+1} - f_{j-1}}{2h} - \underbrace{\frac{f'''(\xi)}{6} h^2}_{E_{\text{Trunck}}}$$

$$\boxed{f_j' = \tilde{f}_j' - E_{\text{Trunck}}}$$

Computer approximation:

$$\hat{f}_j' = \tilde{f}_j' + \varepsilon_j$$

Roundoff error

$$\boxed{\tilde{f}_j' = f_j' + E_{Trunc}}$$

$$\hat{f}_j' = f_j' + E_{Trunc} + \varepsilon_j$$

$$\hat{f}_j' - f_j' = E_{Trunc} + \varepsilon_j$$

$$|\hat{f}_j' - f_j'| \le |E_{Trunc}| + |\varepsilon_j|$$

$\because |a+b| \le |a| + |b|$

Let $|f'''(\xi)| < M$ & $|\varepsilon_j| < \varepsilon$

$$\Rightarrow |\hat{f}_j' - f_j'| \le \frac{Mh^2}{6} + \varepsilon \quad \text{— (2)}$$

$\varepsilon = ?$ It needs to be calculated!

$$\hat{f}_j' = \tilde{f}_j' + \varepsilon_j$$

$$\frac{\hat{f}_{j+1} - \hat{f}_{j-1}}{2h} = \frac{\tilde{f}_{j+1} - \tilde{f}_{j-1}}{2h} + \underbrace{\frac{\varepsilon_{j+1} - \varepsilon_{j-1}}{2h}}$$

$$\Rightarrow \quad \varepsilon = \frac{\varepsilon_{j+1} - \varepsilon_{j-1}}{2h} .$$

Now assume that $\left| \dfrac{\varepsilon_{j+1} - \varepsilon_{j-1}}{2} \right| \leq \underline{c}$

$$\Rightarrow \quad \varepsilon = \frac{c}{h}$$

Use ε in ② to get

$$\Rightarrow \left| \hat{f}_j' - f_j' \right| \leq \frac{M h^2}{6} + \frac{c}{h}.$$

Remarks:

- To reduce truncation error $h \to 0$, but
  - $E_{Round} \longrightarrow \infty$
- Can be minimized! Let $E_{Total} = |\hat{f}' - f_j'|$

$$\frac{\partial E_{Tot}}{\partial h} = 0 \quad \text{has to be satisfied!}$$

$$\text{for} \quad E_{Tot} = \frac{M h^2}{6} + \frac{c}{h}$$

$$\frac{dE_{TOT}}{dh} = \frac{mh}{3} - \frac{c}{h^2} = 0$$

$$\Rightarrow h = \left(\frac{3c}{m}\right)^{1/3}$$

- However, evaluating M and c is almost impossible.

- Higher order approximation is more accurate for a given h. This implies h need not be very small

- Trade-off between E_{Trunc} and E_{Round} is still inherent in numerical differentiation

Numerical Integration:

- Often need to evaluate the definite integral of a function that has no antiderivative or whose antiderivative is hard to obtain.

- The basic method involved in approximating

$\int_a^b f(x)\,dx$ is called "numerical quadrature".

It has the following form:

$$\int_{x=a}^{x=b} f(x)\,dx \simeq \sum_{i=0}^{n} a_i \, \underline{\underline{f(x_i)}}$$

Where a_i - weights of the Quadrature formula.

x_i - quadrature point.

n - no of quadrature pts

Numerical quadrature is based on Interpolating Polynomial.

$$f(x) \simeq \sum_{i=0}^{n} f_i \, L_{N,i}(x)$$

Lagrange Interpolating Polynomial.

$$\int_a^b f(x)\,dx \simeq \sum_{i=0}^{n} f_i \underbrace{\int_a^b L_{N,i}(x)\,dx}_{a_i}$$

$$\Rightarrow \boxed{\int_a^b f(x)\,dx \simeq \sum_{i=0}^{n} f_i \, a_i}$$

$\Rightarrow$ The integration is written as a sum of weighted function value and it is called "Quadrature approximation"

Alternate way is use Taylor approximation!

We derive our formulas using Taylor series.

Let $$I(x) = \int_{t=a}^{t=x} f(t)\, dt$$

where $x \in [a, b]$

Expand $I(x)$ around "a"

$$I(x) = \underset{0}{\cancel{I(a)}} + (x-a) I'(a) + \frac{(x-a)^2}{2!} I''(a) + \cdots$$

Recall: Fundamental Theorem of Calculus

$$I(x) = \int_{a}^{x} f(t)\, dt \quad \forall\, t \in [a, b]$$

$$\Rightarrow \boxed{I'(x) = f(x)}$$

Let $x = b$, then we have

$$\boxed{I(b) = 0 + f(a)h + \frac{f'(a)h^2}{2!} + \frac{f''(a)h^3}{3!} + \cdots}$$

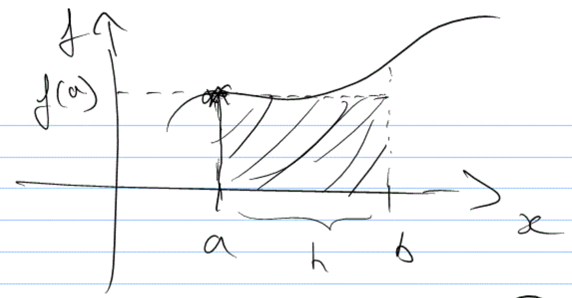
(4)

(i) Simplest (lowest order) Formula

$$I(b) = \int_a^b f(t)\,dt = f(a)h + O(h^2)$$

$$\Rightarrow \quad I(b) \simeq f(a)h.$$

Eg:



(ii) Keep another term in the Taylor expansion (4)

$$\Rightarrow \quad I(b) = \underline{\underline{h f_a}} + \frac{h^2 f_a'}{2} + \frac{h^3}{3!} f_a''(\xi)$$

Truncat. error

Now, use a forward difference approximation for f'

i.e. $f'_a = \dfrac{f_b - f_a}{h} - \dfrac{h}{2} f''(\xi)$

Substitute it to get

$$I(b) = h f_a + \frac{h^2}{2}\left[\frac{f_b - f_a}{h} - \frac{h}{2} f''(\xi_1)\right] + \frac{h^3}{3!} f''(\xi_2)$$

$$= h f_a + \frac{h}{2}(f_b - f_a) + h^3\left[\frac{f''(\tilde{\xi})}{6} - \frac{1}{4} f''(\hat{\xi})\right]$$

$$\Rightarrow \int_a^b f(x)\,dx = \frac{h}{2}\left[f_a + f_b\right] + C h^3 f''(\xi)$$

$$\boxed{\int_a^b f(x)\,dx \simeq \frac{h}{2}\left[f(a) + f(b)\right]}$$

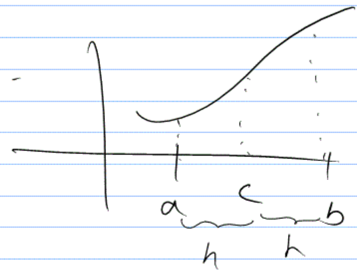
This is Trapezoidal Rule.

Next try the central difference for f' & f''

Expand $I(x)$ around "c"

$$I(x) = I(c) + (x-c)I'(c) + \frac{(x-c)^2}{2!}I''(c) + \cdots$$

$$= I(c) + (x-c)f_c + \frac{(x-c)^2}{2!}f'_c + \cdots$$



Let $x = b$

$$I(b) = I(c) + hf_c + \frac{h^2}{2}f'_c + \frac{h^3}{3!}f''_c + \frac{h^4}{4!}f'''_c + \frac{h^5}{5!}f^{iv}(\xi_b) \quad (1)$$

Let $x = a$

$$I(a) = I(c) - hf_c + \frac{h^2}{2}f'_c - \frac{h^3}{3!}f''_c + \frac{h^4}{4!}f'''_c - \frac{h^5}{5!}f^{iv}(\xi_a) \quad (2)$$

(1) − (2)

$$I(b) = 0 + 2hf_c + \frac{2h^3}{3!}f''_c + \frac{h^5}{5!}\left[f^{iv}(\xi_b) + f^{iv}(\xi_a)\right]$$

use $$f''_c = \frac{f_a - 2f_c + f_b}{h^2} - \frac{h^2}{12}f^{iv}(\xi)$$

$$\Rightarrow \quad I(6) = 2hf_c + \frac{1}{3}h\left[f_a - 2f_c + f_b\right] + h^5\left[c_1\left(f^{iv}(\xi_b) + f^{iv}(\xi_a)\right) - c_2 f^{iv}(\xi)\right]$$

$$I(6) = h\left[6f_c + \frac{f_a - 2f_c + f_b}{3}\right] + O(h^5)$$

$$\Rightarrow \quad \boxed{\int_a^b f(x)\,dx = \frac{h}{3}\left[f_a + 4f_c + f_b\right] + O(h^5)}$$

Simpson's 1/3 Rule.

Some of the Common Closed Newton-Cotes formulae:

(1) Trapezoidal Rule:

$$\int_{x_0}^{x_1} f(x)\,dx = \frac{h}{2}\left[f_0 + f_1\right] - \frac{h^3}{12}f''(\xi)$$

where $f_i = f(x_i)$ & $\xi \in (x_0, x_1)$

(2) Simpson's Rule:

$$\int_{x_0}^{x_2} f(x)\,dx = \frac{h}{3}\left[f_0 + 4f_1 + f_2\right] - \frac{h^5}{90}f^{4}(\xi)$$

③ Simpson 3/8 Rule:

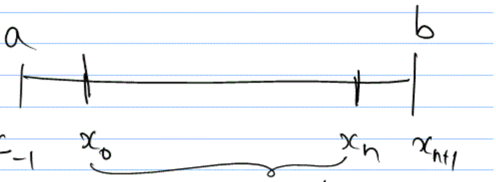
$$\int_{x_0}^{x_3} f(x)\,dx = \frac{3h}{8}\left[f_0 + 3f_1 + 3f_2 + f_3\right] - \frac{3h^5}{80} f^4(\xi)\,!$$

④ Boole's Rule:

$$\int_{x_0}^{x_4} f(x)\,dx = \frac{2h}{45}\left[7f_0 + 32f_1 + 12f_2 + 32f_3 + 7f_4\right] - \frac{8h^7}{945} f^6(\xi)$$

Open Newton-Cotes Formulas:

Let $h = \dfrac{b-a}{n+2}$



define : $x_i = x_0 + ih, \; i = 0, \dots n$ / $n+2 = 1 + \underbrace{\quad}_{n\text{-intervals}} + 1$

with $x_0 = a + h$.

$\Rightarrow \quad x_n = x_0 + ih = b - h.$

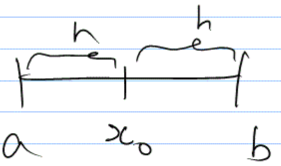
Now, the sample points in the open formula will be in the open interval (a, b)

ie $\displaystyle\int_a^b f(x)\,dx = \int_{x_{-1}}^{x_{n+1}} f(x)\,dx \simeq \sum_{i=0}^{n} w_i f(x_i)$

where $\displaystyle w_i = \int_a^b L_i(x)\,dx$ (weights)

<u>Mid-point Rule:</u> (open Newton Cotes Formula)

<u>$n=0$:</u>



$$I(x) = I(x_0) + f_0(x-x_0) + \frac{f_0'}{2}(x-x_0)^2 + \dots$$

Let $x=b$

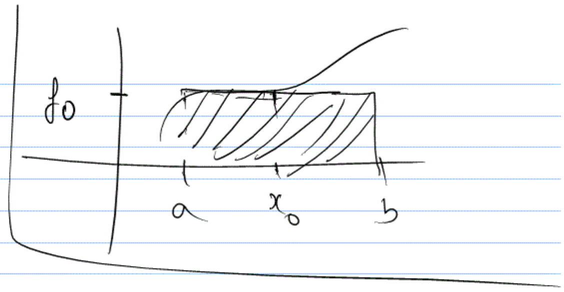
$$I(b) = I(x_0) + hf_0 + \frac{h^2}{2}f_0' + \frac{h^3}{3!}f''(\xi_b) \quad \text{— (1)}$$

$$I(a) = I(x_0) - hf_0 + \frac{h^2}{2}f_0' - \frac{h^3}{3!}f''(\xi_a) \quad \text{— (2)}$$

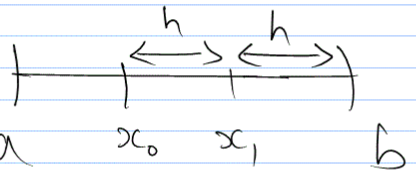
(1) − (2) ⇒

$$I(b) = 2hf_0 + \frac{h^3}{3!}\left[f''(\xi_b) + f''(\xi_a)\right]$$

$$\Rightarrow \boxed{\int_a^b f(x)\,dx = 2hf_0 + \frac{h^3}{3!}f''(\xi)}$$



(ii) $n = 2$



$$I(b) = I(x_0) + 2h f_0 + \frac{(2h)^2}{2} f_0' + \frac{(2h)^3}{3!} f''(\xi_b) \cdots \quad ①$$

$$I(a) = I(x_0) - h f_0 + \frac{h^2}{2} f_0' - \frac{h^3}{3!} f''(\xi_a) \cdots \quad ②$$

① − ②

$$I(b) = \frac{3h}{2} \left[f_0 + f_1 \right] + \frac{3}{2} h^3 f''(\xi)$$

$$\Rightarrow \boxed{\int_a^b f(x)\,dx \simeq \frac{3h}{2} \left[f_0 + f_1 \right]}$$

Common open Newtons-Cotes Formulae:

$n=0$: Midpoint rule:

$$\int_{x_1}^{x_1} f(x)\,dx = 2h f(x_0) + \frac{h^3}{3} f''(\xi)$$

$n=1$:

$$\int_{x_{-1}}^{x_2} f(x)\,dx = \frac{3h}{2}\left[f_0 + f_1\right] + \frac{3h^3}{4} f''(\xi)$$

$n=2$

$$\int_{x_{-1}}^{x_3} f(x)\,dx = \frac{4h}{3}\left[2f_0 - f_1 + 2f_2\right] + \frac{14h^5}{45} f^4(\xi)$$

$n=3$:

$$\int_{x_{-1}}^{x_4} f(x)\,dx = \frac{5h}{24}\left[11f_0 + f_1 + f_2 + 11f_3\right] + \frac{95}{144} h^5 f^4(\xi)$$

Composite Numerical Integration

— Newton-cotes formulaes are generally unsuitable for use over large integration intervals

— Higher-degree formulas would be required.

— Newton-Cotes formulaes are based on interpolating polynomials that use equally spaced points, which will be inaccurate over large invertvals for oscillating high-degree polynomials.

$\Longrightarrow$ Approximate the interval piecewise

Ex: Let $f(x) = e^x$ evaluate

$\int_0^4 e^x dx$ using two subintervaland Simpson's $1/3$ rule.

$$\int_0^4 e^x dx = \int_0^2 e^x dx + \int_2^4 e^x dx$$

$= 53.86$ S2 (using Simpson's Rule)

$$h=4 \Rightarrow \int_0^4 e^x\,dx \simeq \frac{2}{3}\left(e^0+4e^2+e^4\right) \text{ (Simpson)}$$

use one interval

$$= 56.76 \leftarrow S1$$

$$h=1 \Rightarrow \int_0^4 e^x\,dx = \int_0^1 + \int_1^2 + \int_2^3 + \int_3^4$$

using 4 subinterval

$$= 53.61 \text{ — } S3$$

$$\boxed{\int_0^4 e^x\,dx = 53.59} \text{ — } E$$

(use 1 interval) $e_1 = E - S1 = -3.17143$

$$e_2 = E - S_2 = -0.2657$$

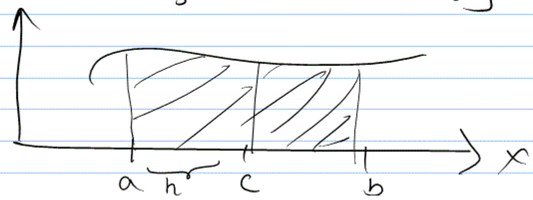
$$e_3 = E - S_3 = -0.01807$$

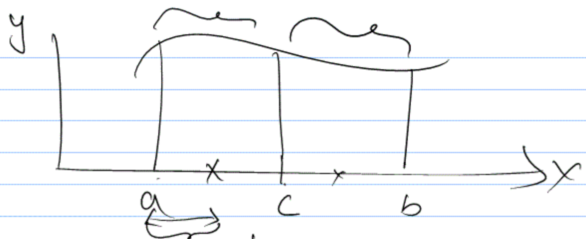
Adaptive Quadrature for Simpson's Rule:

Simpson's Rule:

$$\int_a^b f(x)\,dx = \frac{h}{3}\left[f_a + 4 f_c + f_b\right] - \frac{h^5 f^4(\xi)}{90}$$

Let $S(a,b) = \frac{h}{3}\left[f_a + 4f_c + f_b\right]$ ——— (1)





$h_1 = \frac{h}{2}$

$$\int_a^b f(x)\,dx = S(a,c) - \frac{h_1^5 f^4(\xi_1)}{90}$$

$$+ S(c,b) - \frac{h_1^5 f^4(\xi_2)}{90}$$

Assume $\bar{f}^4(\xi) = \dfrac{f^4(\xi_1) + f^4(\xi_2)}{2}$

$$\int_a^b f(x)\,dx = S(a,c) + S(c,b) - \frac{2\left(h/2\right)^5 \bar{f}^4(\xi)}{90}$$

$$= S(a,c) + S(c,b) - \frac{1}{16}\left[\frac{h^5 \bar{f}^4(\xi)}{90}\right] \quad (2)$$

Now equate eqn (1) & (2), to get

$$S(a,b) - \frac{h^5 f^4(\xi)}{90} = S(a,c) + S(c,b) - \frac{1}{16}\left[\frac{h^5 \bar{f}^4(\xi)}{90}\right]$$

Assume that $f^4(\xi) = \bar{f}^4(\xi)$

$$\Rightarrow \quad S(a,b) - S(a,c) - S(c,b) = \frac{15}{16}\left[\frac{h^5 \bar{f}^4(\xi)}{90}\right]$$

$$\Rightarrow \quad \frac{1}{16}\left[\frac{h^5 \bar{f}^4(\xi)}{90}\right] = \frac{1}{15}\left[S(a,b) - S(a,c) - S(c,b)\right]$$

use 16 in (2) to get

$$\left|\int_a^b f(x)\,dx - S(a,c) - S(c,b)\right| = \frac{1}{15}\left|S(a,b) - S(a,c) - S(c,b)\right|$$

Remark:

- $S(a,c) + S(c,b)$ approximates $\int_a^b f(x)\,dx$ 15-times better than it agrees with the known value $S(a,b)$

- Each successive application of Simpson's rule on a interval $[a,b]$ with $h_{i+1} = \frac{h_i}{2}$, $i = 0, 1, \dots$ where $h_0 = h$, leads roughly 15 times more accurate.

- The assumption $f^4(\xi) = \tilde{f}^4(\xi)$ is used, thus to compensate the inequality often more conservative factor (10) is used.

How to use it:

$a \vdash\!\!\overset{S_{10} = S(a,b)}{\text{———}}\!\!\dashv b$ — Level 0

$S_{11} = S\left(a, \frac{a+b}{2}\right)$ $\quad S_{2,1} = S\left(\frac{a+b}{2}, b\right)$

a ———————— b — Level 1

$S_{2,1}$: 2 = interval, 1 = refinement level

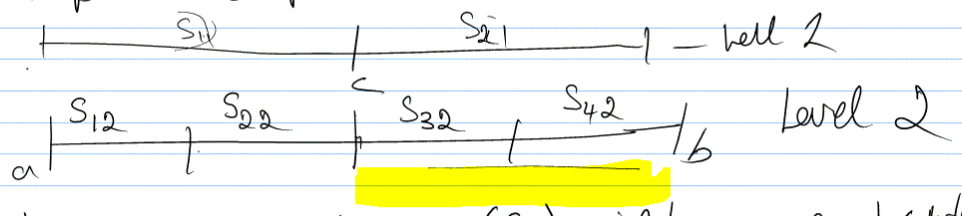
if $\underline{|S_{10} - S_{11} - S_{21}| < 15\,\varepsilon}$

$$\Rightarrow \left| \int_a^b f(x)\,dx - S_{11} - S_{21} \right| < \varepsilon$$

needed accuracy

Thus stop the computation

else

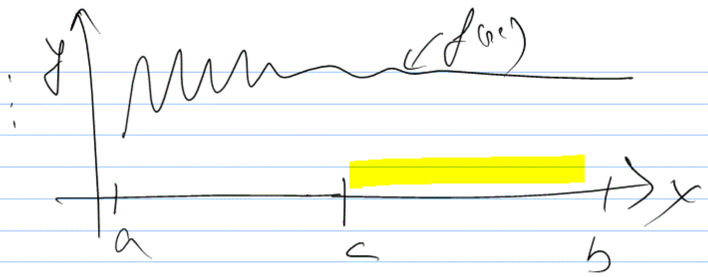


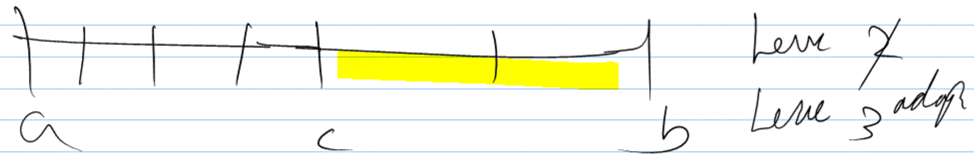
if $|S_{11} - S_{12} - S_{22}| < 15(\varepsilon/2)$ if $|S_{21} - S_{32} - S_{42}| < 15(\varepsilon/2)$

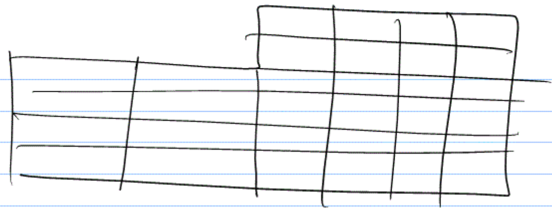
$\Rightarrow$ stop.

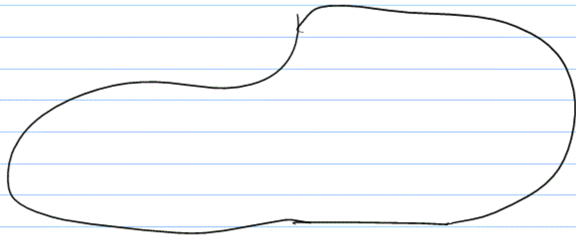
$\Rightarrow \left| \int_a^c f(x) - S_{12} - S_{22} \right| < \frac{\varepsilon}{2}$; $\left| \int_c^b f(x)\,dx - S_{32} - S_{42} \right| < \varepsilon/2$

else









Gaussian Quadrature:

Is there an optimal set of sample points for approximating $\int_a^b f(x)\,dx$?

— optimal in the sense that we can integrate exactly the maximum degree of polynomial using the minimum number of sample points.

Recall $\int_a^b f(x)\,dx \simeq \sum_{i=1}^{N} a_i f(x_i)$

- There are special sample points $\{x_i\}$ called Gaussian points which are the roots of Legendre polynomial.

- By using these points, we can exactly integrate a polynomial of degree $2N-1$, where N is the number of sample points.

Ex: Let $N=2$;

Eg: Trapezoidal rule:

$$\int_a^b f(x)\,dx = I(b) = \frac{h}{2}\left[f_a + f_b\right] - \frac{h^3}{12} f''(\xi)$$

- Could do only a linear functions exactly.

- But, we should be able to integrate upto cubic polynomials exactly ($\because 2N-1 = 4-1$)

Consider the Case $N=2$.

As per the rule, we should be able to integrate a polynomial of degree three exactly.

ie

$$\int_{-1}^{1} f(x)\,dx \simeq c_1 f(x_1) + c_2 f(x_2)$$

Where

c_1, c_2 are weights

x_1, x_2 are Quadrature point

and $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$

$$\int_{-1}^{1} f(x)\,dx = \underbrace{\int_{-1}^{1} a_0\,dx}_{I_1} + \underbrace{\int_{-1}^{1} a_1 x}_{I_2} + \underbrace{\int_{-1}^{1} a_2 x^2}_{I_3} + \underbrace{\int_{-1}^{1} a_3 x^3}_{I_4}$$

$\Rightarrow$ equivalent to show that the formula gives exact result for $f(x)$.

Consider I_1, ie $f(x) = 1$;

$\Rightarrow$

$$I_1 = a_0 \underbrace{\int_{-1}^{1} dx}$$

$$= \cancel{a_0}(c_1 + c_2) = \cancel{a_0}\,2$$

$$\cancel{a_0}\,(C_1 + C_2) = a_0 \int_{-1}^{1} 1 \cdot dx = 2\cancel{a_0} \quad \text{——} \; (1)$$

For I_2, $f(x) = x$

$$c_1 x_1 + c_2 x_2 = 0 \quad \text{——} \; (2)$$

For I_3:

$$f(x) = x^2$$

$$I_3 = a_2 \int_{-1}^{1} x^2 \, dx$$

$$\cancel{a_2}\left[c_1 x_1^2 + c_2 x_2^2\right] = \cancel{a_2}\,\frac{2}{3} \quad \text{——} \; (3)$$

I_4:

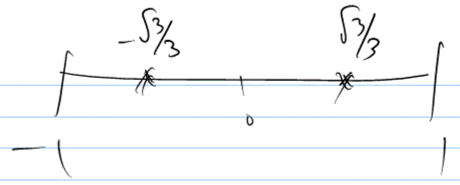
$$c_1 x_1^3 + c_2 x_2^3 = 0 \quad \text{——} \; (4)$$

Simple evaluation gives. [Exercise]

$$c_1 = 1, \quad c_2 = 1, \quad x_1 = -\frac{\sqrt{3}}{3}, \quad x_2 = \frac{\sqrt{3}}{3}$$

i.e. $$\int_{-1}^{1} f(x)\,dx \simeq f\left(\frac{-\sqrt{3}}{3}\right) + f\left(\frac{\sqrt{3}}{3}\right)$$

which is Gaussian Quadrature formula for $n=2$ in one Dimension.



Remarks:

- This technique can be used to identify the weights and Quadrature points that give exact results for higer-degree polynomial.

- Alternatively, we can obtain more easily, by using roots of Legendre polynomial.

- First few Leagendre polynomials:

$$P_0(x) = 1, \quad P_1(x) = x, \quad P_2(x) = x^2 - \frac{1}{3}$$

$$P_3(x) = x^3 - \frac{3}{5}x, \quad P_4(x) = x^4 - \frac{6}{7}x^2 + \frac{3}{35}$$

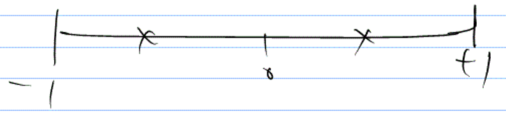
- The roots of Legendre polynomial lie in the interval $(-1, 1)$, and have symmetry w.r.t the origin

- See table 4.11, p.No: 215 of the text book for Quadrature points and weights

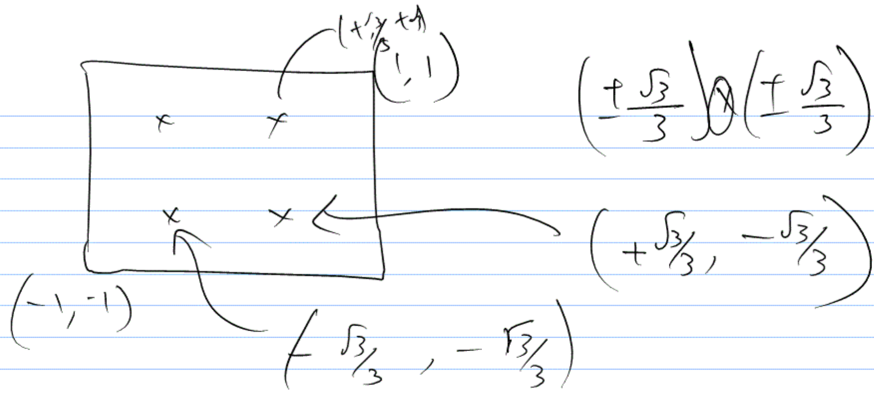
How to construct high dimensional formulas?

Ans: Take tensor product of 1D pts.

1D

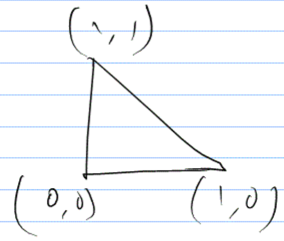


2D



$$\Rightarrow \int_{-1}^{1}\int_{-1}^{1} f(x,y)\,dx\,dy \simeq \sum_{i=1}^{4} w_i f(x_i, y_i)$$

— The Gaussian Quadrature formulas are most commonly used in practice.



Initial value problem for ODE

Ordinary differential equations – derivatives w.r.t one independent variable.

Partial differential equations – derivatives w.r.t more one independent variable.

Ex: of ODE

$$y'(x) = f(y, x)$$

— first order

n^{th} order ODE:

$$\frac{d^n y}{dx^n} = f\left(x, y, \frac{dy}{dx}, \frac{d^2 y}{dx^2}, \cdots, \frac{d^{n-1} y}{dx^{n-1}}\right)$$

General form of n^{th} order linear ODE:

$$a_n(x)\frac{d^n y}{dx^n} + a_{n-1}(x)\frac{d^{n-1} y}{dx^{n-1}} + \cdots + a_0(x)\, y = f(x).$$

- linear, ie no power on y
- $f(x) = 0 \Rightarrow$ homogeneous equation

Eg:

① $\dfrac{dy}{dx} - y\cos x = \sin x$

1st order linear ODE.

② $\dfrac{d^4 y}{dx^4} = 6y^3$

4th order nonlinear ODE.

Eg. of PDEs:

$$\frac{\partial^2 u(x,t)}{\partial x^2} - \frac{\partial^2 u(x,t)}{\partial t^2} = 0 \qquad \text{(wave equation)}$$

$$K\, u_{xx}(x,t) - u_t(x,t) = 0 \qquad \text{(energy equation)}$$

$$u_{xx}(x,y) + u_{yy}(x,y) = 0 \qquad \text{(Laplace eqn)}$$

Solution to First order ODE:

$$\underbrace{y'(t)}_{\text{derivative}} = \underbrace{f(y,t)}_{\text{rate function}} \quad ; \quad \underbrace{y(t_0) = y_0}_{\text{Initial Condition.}}$$

- We want to find $y(t)$ for given rate function $f(y,t)$ and the initial Condition.
- Let us Consider the Simple case:

$f(y,t) = f(t)$ i.e. rate fn is only a fn of "t."

$$\Rightarrow \quad y'(t) = f(t)$$

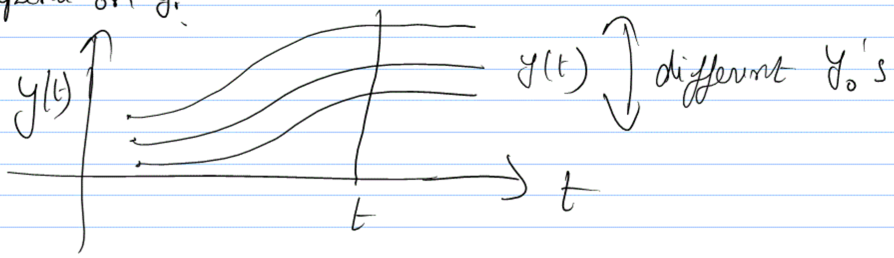
$$y(t) = \int_{t'=t_0}^{t'=t} f(t')\,dt' + y_0$$

G. S

$$\int dy = \int f(x)\,dx$$

$$\Rightarrow y = F(x) + C$$

- The Solution set is a family of curves with a constant slope (rate function at a fixed value of t). ie $y'(t) = f(t)$ does not depend on y.



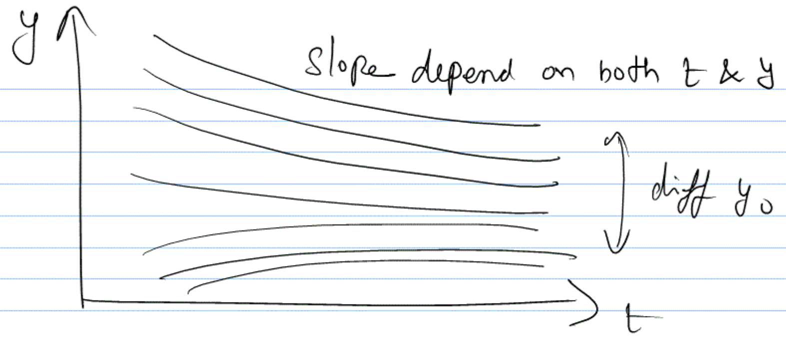
- Let us consider the more general case:

$f = f(y,t)$ ie f is a fun of "y" & "t"

$$\Rightarrow \quad y'(t) = f(y,t)$$

$$y(t) = \int_{t_0}^{t} f(y,t')\,dt' + y_0$$

Eg:



Ex: Solve

$$\frac{dy}{dt} = \cos t + \sin t, \qquad y(\pi) = 1.$$

$$dy = (\cos t + \sin t)\, dt$$

$$\Rightarrow y = \sin t - \cos t + C$$

when $t = \pi$ $y = 1$

$$\Rightarrow 1 = \sin\pi - \cos\pi + C$$

$$\Rightarrow C = 0$$

$$\boxed{y(t) = \sin t - \cos t}$$

Euler's method for ODE

- Rarely used in practice (not very accurate) but good to demonstrate principles to solve IVPs.
- three different ways to think about approximating solutions of ODEs.

(i) Finite difference approximation.

$$y' = \frac{y_{i+1} - y_i}{h} - \frac{h}{2} y''(y(\xi_i), \xi_i) \quad (1)$$

note $h = t_{i+1} - t_i$ & $y'' = f'$ ($\because y' = f$)

using (1) in our model, we get

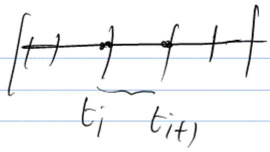
$$\Rightarrow \frac{y_{i+1} - y_i}{h} - \frac{h}{2} y''(y(\xi_i), \xi_i) = f(y_i, t_i)$$

$$\Rightarrow y_{i+1} = y_i + h f_i + \frac{h^2}{2} f'(y(\xi_i), \xi_i)$$

Let $w_i \simeq y_i$ & truncate the error term to get

$$\boxed{w_{i+1} = w_i + h f_i}, \quad i = 0, \cdots n$$

(ii) Integrate the ODE



$$\int_{t'=t_i}^{t_{i+1}} y'\, dt' = \int_{t'=t_i}^{t'=t_{i+1}} f(y_i, t')\, dt' = I(t)$$

$$y_{i+1} - y_i = I(t_i) + (t-t_i) f_i + \frac{(t-t_i)^2}{2!} f'(\xi(t), \xi_i)$$

$$= 0 + h f_i + \frac{h^2}{2!} f'(\xi) \qquad (t = t_{i+1})$$

$$\Rightarrow \boxed{y_{i+1} = y_i + h f_i + \frac{h^2}{2!} f'(\xi)}$$

Which is same as the finite diff.

(iii) Taylor series expansion of $y(t)$ around t_i and evaluate at t_{i+1}

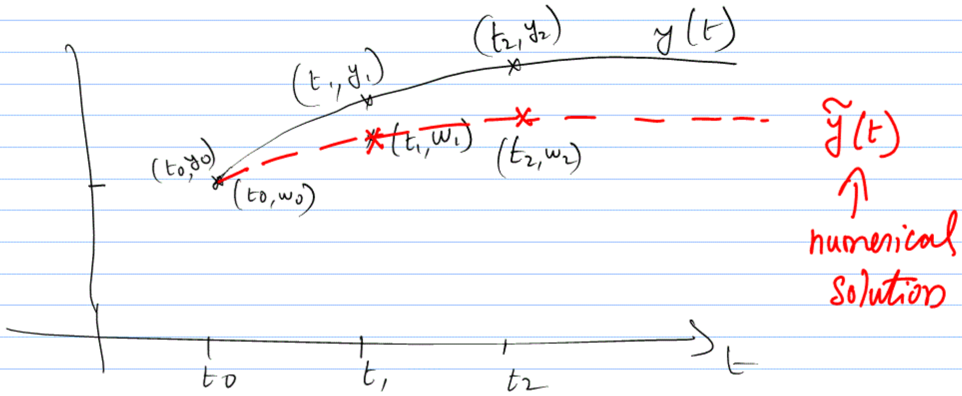
$$y(t) = y(t_i) + y'(t_i)(t-t_i) + \frac{y''(\xi)}{2}(t-t_i)^2$$

$$y(t_{i+1}) = y(t_i) + \underbrace{y'(t_i)} h + \frac{y''(\xi)}{2} h^2$$

$$\Rightarrow \boxed{y_{i+1} = y_i + h f_i + \frac{h^2}{2} f'(\xi)}$$

All these three ways give

$$\boxed{w_{i+1} = w_i + h f_i} \quad \text{Euler's method.}$$



Remarks:

- It is OK if $f(y,t) = f(t)$, otherwise it induce additional error since it has to be evaluated at $f(y_i, t_i)$ not at $f(w_i, t_i)$

- Euler method is an explicit method
- new solution is a function of old solution
- local truncation error is $O(h^2)$

- $(y_{i+1} - y_i)$ is $O(h)$ as t advances
- additional error due to evaluation of the rate function at (w_i, t_i) instead of (y_i, t_i) induce more error.

Let us look this error component more quantitatively.

Define $\varepsilon_{i+1} = |y_{i+1} - w_{i+1}|$

Standard analysis: $\Rightarrow$

$$y_{i+1} = y_i + h f(y_i, t_i) + \frac{h^2}{2} f'(y(\xi_i), \xi_i) \quad -(1)$$
$$w_{i+1} = w_i + h f(w_i, t_i) \quad -(2)$$

denote $f_i = f(y_i, t_i)$ & $\bar{f}_i = f(w_i, t_i)$

From (1) - (2)

$$\varepsilon_{i+1} = \varepsilon_i + h(f_i - \bar{f}_i) + \frac{h^2}{2} f'(y(\xi_i), \xi_i)$$

$$= \varepsilon_i \left[1 + h \frac{(f_i - \bar{f}_i)}{\underbrace{y_i - w_i}}\right] + \frac{h^2}{2} f'(y(\xi_i), \xi_i)$$

By mean value thorem $f'(c) = \dfrac{f(b) - f(a)}{b - a}$ $\in (a,b)$

$$\Rightarrow \frac{f_i - \bar{f}_i}{y_i - w_i} = \underbrace{\frac{\partial f(\eta, t_i)}{\partial y}}_{J_i}, \text{ where } \eta \in (y_i, w_i)$$

J_i — Jacobian of the eqn at "t_i"

$$\Rightarrow \boxed{\mathcal{E}_{i+1} = \mathcal{E}_i \underbrace{(1 + h J_i)}_{\text{amplification factor}} + \underbrace{\frac{h^2}{2} f'(y(\xi_i), \xi_i)}_{\text{truncation error}}}$$

Remarks:

- error propogates as time advances.
- wants J_i small and negative, but it is not the case.
- For Euler method we need:

$$|1 + h J_i| < 1$$

otherwise global error blows up

$$\Rightarrow \quad -1 < 1 + h J_i < 1$$

$|a| < 1$
$\Rightarrow -1 < a < 1$

$$-2 < hJ_i < 0$$

$\Rightarrow$ J_i must be less than zero.

otherwise

$\Rightarrow$ $0 < 2 + hJ_i < 2$

$\Rightarrow$ $2 + hJ_i > 0$

$\Rightarrow$ $h > \dfrac{2}{|J_i|}$

$J_i < 0$

$J_i = -|J_i|$

which is the stability condition for the method and limitation on the mesh size.

Higher order Taylors method:

Let $w_0 = \alpha$ [initial guess]

$$w_{i+1} = w_i + h\,T^n(t_i, w_i), \quad \forall\, i = 0, 1, \ldots n-1,$$

where
$$T^n(t_i, w_i) = f(t_i, w_i) + \frac{h}{2} f'(t_i, w_i) + \cdots + \cdots + \frac{h^{n-1}}{n!} f^{n-1}(t_i, w_i)$$

$\boxed{\text{Recall: } y'(t) = f(y, t) = f(y(t), t)}$

Remarks:

For $n=1$ $\Rightarrow$ Eulers method ($O(h)$ Globally)

$n=2$ $\Rightarrow$ Taylor's method with [$O(h^2)$ Globally]

w.r.t $f = f(y(t), t) = y'(t)$

$$f' = \frac{df}{dt} + \frac{df}{dy}\frac{dy}{dt}$$

$$\boxed{f' = \frac{df(y(t),t)}{dt} + \frac{df}{dy} f}$$

$n=3$ $\Rightarrow$ Taylor's method of order 3. (globally)

$$f'' = \left(\frac{df}{dt}\right)' + \left(\frac{df}{dy}\right)' f + \quad \text{(Exercise}$$

- Higher order Taylor methods quickly become tedious
- needs evaluation of derivatives of f'', f'.

Runge-Kutta methods

- Approximate the derivatives using weighted function evaluations of "f".
- Restrict evaluations to $[t_i, t_{i+1}]$

$\Longrightarrow$ RK methods:

RK 2

$$w_{i+1} = w_i + h f_i + \frac{h^2}{2} \underbrace{\left(\frac{df_i}{dt} + \frac{df_i}{dy} f_i \right)}_{\text{Replace with function evaluation.}} \quad \text{— (1)}$$

For which, we need 2D Taylor expansion.

Expand $f(y, t)$ around (w_i, t_i).

$$f(y,t) = f(w_i, t_i) + \left[(t - t_i) \frac{df_i}{dt} + (w - w_i) \frac{df_i}{dy} \right] + \left[(t - t_i)^2 \frac{d^2 f_i}{dt^2} + (w - w_i)^2 \frac{d^2 f_i}{dy^2} \right] + \text{Error}$$

where $f_i = f(w_i, t_i)$

Recall IVP

$$y' = f(y,t), \quad y(t_0) = y_0.$$

$$y_{i+1} = y_i + h f_i + \frac{h^2}{2} f_i' + \cdots \quad \text{--- (2)}$$

use $\boxed{f' = \frac{d}{dt}\left(f(y,t)\right) = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} \cdot f}$

Let $t = t_i + \beta$, $y = w_i + \alpha$ in 2D Taylor form

$$\Rightarrow f(w_i+\alpha, t_i+\beta) = f_i + \beta \frac{\partial f_i}{\partial t} + \alpha \frac{\partial f_i}{\partial y} + \frac{\beta^2}{2} \frac{\partial^2 f_i}{\partial t^2} + \frac{\alpha^2}{2} \frac{\partial^2 f_i}{\partial y^2} + \alpha\beta \frac{\partial^2 f_i}{\partial t \partial y} + \cdots$$

Now consider up to first derivative in α & β

$$\Rightarrow \boxed{f(w_i+\alpha, t_i+\beta) = f_i + \beta \frac{\partial f_i}{\partial t} + \alpha \frac{\partial f_i}{\partial y}} \quad \text{--- (3)}$$

Now rewriting the original expression (1) to get

$$w_{i+1} = w_i + h\left[f_i + \overbrace{\frac{h}{2}}^{\beta}\frac{df_i}{dt} + \overbrace{\frac{h}{2}\frac{df}{dy}f_i}^{\alpha}\right]$$

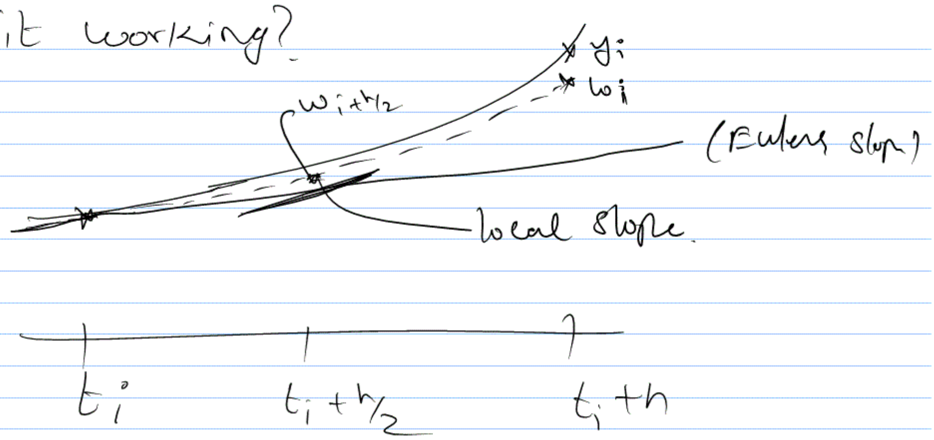
using (3) and taking $\beta = \frac{h}{2}$ and $\alpha = \frac{h}{2} f_i$

we get

$$\Rightarrow \boxed{w_{i+1} = w_i + h f\left(w_i + \frac{h}{2} f_i,\ t_i + \frac{h}{2}\right)}$$

$\Rightarrow$ mid point Rule!

How is it working?



Note that the midpoint rule is a particular choice of a more general approximation to

$$f_i + \frac{h}{2}\left(\frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} f\right) = a_1 f(w_i, t_i) + a_2 f(w_i + \beta, t_i + \alpha)$$

ie $a_1 = 0$, $a_2 = 1$, $\alpha = \frac{h}{2} f_i$, $\beta = \frac{h}{2}$ in midpoint rule!

— Another choice leads to the "modified Eulers method" (or) "Trapezoidal rule"

$$w_{i+1} = w_i + \frac{h}{2}\left[f(w_i, t_i) + f\left(w_i + h f(w_i, t_i), t_{i+1}\right)\right]$$

here $\alpha = h$, $\beta = h f_i$, $a_1 = a_2 = \frac{1}{2}$

— Another choice leads to Heun's method!

$$w_{i+1} = w_i + \frac{h}{4}\left[f(w_i, t_i) + 3 f\left(w_i + \frac{2}{3} h f_i, t_i + \frac{2}{3} h\right)\right]$$

Here $a_1 = \frac{1}{4}$, $a_2 = \frac{3}{4}$

$\alpha = \frac{2}{3} h$, $\beta = \frac{2}{3} h f_i$

— The above two methods are of second order and classified as RK2

- $O(h^3)$ RK methods are not generally used!

- The most commonly used (also very popular) RK method is the RK4. (ie 4th order RK method)

- RK4 can be derived from integration of the ODE using Simpson's Rule (instead of Trapezoidal in RK2)

$$\int_{t_i}^{t_{i+1}} y' \, dt = \int_{t_i}^{t_{i+1}} f(y(t),t) \, dt = \frac{h}{6}\left[f_i + 4 f_{i+\frac{1}{2}} + f_{i+1}\right] + O(h^3)$$

Exercise

$$\Rightarrow \quad w_{i+1} = w_i + \frac{h}{6}\left[f_i + 4 f_{i+1/2} + f_{i+1}\right]$$

Note that in the above formula, we still need to evaluate f at $i+1/2$ & $i+1$.

$$\boxed{\text{where } f_{i+1/2} = f(x_i + h/2, \ldots)}$$

– There are many possibilities to evaluate f at $i+1/2$ & $i+1$. However, the most famous form is

$$w_{i+1} = w_i + h\left[\frac{f_1 + 2(f_2 + f_3) + f_4}{6}\right]$$

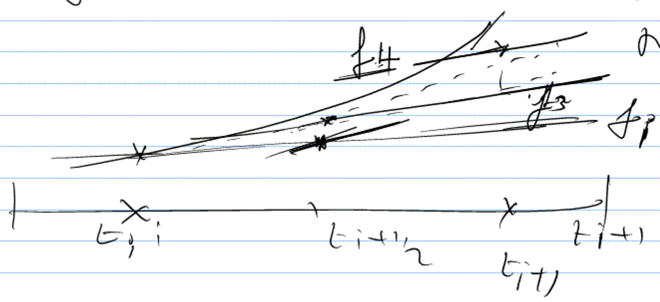
where

$f_1 = f_i$ ← Slope at f_i

$f_2 = f\left(\underline{\underline{w_i + \frac{h}{2} f_1}},\ t_{i+1/2}\right)$ ← slope at midpt using f_1

$f_3 = f\left(w_i + \frac{h f_2}{2},\ t_{i+1/2}\right)$

$f_4 = f\left(w_i + h f_3,\ t_{i+1}\right)$ – Slope at end pt using f_3 to get the update.



Remarks on RK4

- single-step methods at (w_i, t_i) & (w_{i+1}, t_{i+1})
- we donot use any information about the rate function outside the interval.

Multi step method:

- use past information about rate function.
- more risky and likely to become instable

- we will restrict approximations to derivatives to sample on one interval.
- Instead use f_i, f_{i-1}, f_{i-2} to approximate $f', f'', \ldots$
- If we use f_{i+1}, (eg. Trapezoidal method) the we must calculate w_{i+1}!
- In general N-step method will use

n-1 single steps of comparable order.

Class of Multi-step methods:

- **Adams-Bashforth methods:**

  "open" (or) explicit methods, ie do not involve f_{i+1} to evaluate w_{i+1}

- **Adams-moulton methods:**

  "closed" (or) implicit methods, it involves f_{i+1}

in the evaluation of w_{i+1}

Basic idea:

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2} f_i' + \frac{h^3}{3!} f_i'' + \cdots$$

derivative approximate with

$$\underbrace{f_{i+1}}_{\text{implicit}}, \underbrace{f_i, f_{i-1}, f_{i-2}}_{\text{explicit}} \cdots$$

Adams - Bashforth 2-step method

$$y_{i+1} = y_i + h f_i + \frac{h^2}{2} f_i' + \cdots$$

$\because y' = f$

use first order backward difference for "f_i'"

i.e. $$f_i' = \frac{f_i - f_{i-1}}{h} + O(h)$$

$\Rightarrow$ in iterative form

$$w_{i+1} = w_i + h f_i + \frac{h^2}{2}\left[\frac{f_i - f_{i-1}}{h} + \frac{h^2}{2}(O(h))\right]$$

$$\Rightarrow \boxed{w_{i+1} = w_i + \frac{h}{2}\left[3f_i - f_{i-1}\right]}$$

Remarks:

- Same as RK 2, but no half intervals
- Keep track on previous function evaluation and only one function evaluation required per step.

Adams - Bashforth 3- step method:

Use 3 points to approximate

f' to $O(h^2)$ and f'' to $O(h)$

Exercise:

$$\Rightarrow w_{i+1} = w_i + \frac{h}{12}\left[23 f_i - 16 f_{i-1} + 5 f_{i-2}\right]$$

— This scheme is $\underline{\underline{O(h^3)}}$ overall.

How ?

— We can construct by taking more points to get A-B method of 2 — 5 steps.

Adams - Moulton method:

— involves f_{i+1}, so we can use central difference approximation, ie f' to $O(h^2)$ with only 2 - steps.

A-M 2 step method

– apply Central difference to these terms:

$$\frac{h^2}{2!} f_i' = \frac{h^2}{2}\left[\frac{f_{i+1} - f_{i-1}}{2h} + O(h^2)\right]$$

$$\frac{h^3}{3!} f_i'' = \frac{h^3}{3!}\left[\frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + O(h^2)\right]$$

$$\Rightarrow w_{i+1} = w_i + h f_i + \frac{h^2}{2}\left[\frac{f_{i+1} - f_{i-1}}{2h}\right] + \frac{h^3}{3!}\left[\frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}\right]$$

$$\boxed{w_{i+1} = w_i + \frac{h}{12}\left[5 f_{i+1} + 8 f_i - f_{i-1}\right]}$$

A-M 2step is $O(h^3)$ overall

Note:

$f_{i+1} = f(w_{i+1}, t_{i+1})$ is needed in the implicit A-M method

i.e. w_{i+1} is needed to evaluate f_{i+1}

Two ways of handling implicit terms:

1) y appears linear in $f(y,t)$

Eg: $y' = -y + t + 1$

Then f_{i+1} evaluation is not a problem, since using 2 step A-M method

$$w_{i+1} = w_i + \frac{h}{12}\left[5(-w_{i+1} + t_{i+1} + 1) + \cdots\right]$$

$$\Rightarrow \quad w_{i+1}\left[1 + \frac{5h}{12}\right] = w_i + \frac{h}{12}(t_{i+1} + 1) + \cdots$$

2) y appears nonlinearly in $f(y,t)$

Eg: $y' = \cos y + t$

Then f_{i+1} evaluation presents a problem when 2 step A-M is used.

ie $w_{i+1} = w_i + \frac{h}{12}\left[5\left[\cos(w_{i+1}) + t_{i+1}\right] + \cdots\right]$

– not possible to combine w_{i+1} to L.H.S.

– instead, we must predict w_{i+1} and then correct with the method

Predictor / Corrector Schemes:

– use an explicit method to predict w_{i+1} then use the implicit method to correct / improve the predicted value.

– We will derive the scheme for implicit Trapezoidal method in the form of modified Eulers method.

$$w_{i+1} = w_i + \frac{h}{2}\left[f_i + f_{i+1}\right] \quad \text{(Trapezoidal)}$$

where $f_{i+1} = f(\tilde{w}_{i+1}, t_{i+1})$

Replace "w_{i+1}" with $\tilde{w}_{i+1} = w_i + h f_i$ (Euler step) in " f_{i+1} "

Eg:

A-B / A·M P/C methods can be constructed

Predict: $\tilde{w}_{i+1} = w_i + \frac{h}{2}\left(3f_i - f_{i-1}\right)$ AB 2-step.

Correct: $w_{i+1}^{k+1} = w_i + \frac{h}{12}\left[5 f(w_{i+1}^{k}, t_{i+1}) + 8f_i - f_{i-1}\right]$ A-M 2 step

$$\boxed{w_{i+1}^{k+1} = g\left(w_{i+1}^{k}\right)}$$

General Remark:

- use a predictor no more than one order of "h" lower.
- Correction step can be repeated; think of it as a fixed point scheme.

ie $W_{i+1}^{k+1} = g\left(w_{i+1}^{k}\right)$, $k=0,\ldots$ (fixed pt iteration step)

with $w_{i+1}^{0} = \tilde{w}_{i+1}$ (ie from explicit method).

- In most cases there is no advantage of "correcting" more than once, since the iteration converges to w_{i+1} not to y_{i+1}, the exact solution.

- If we do iteration, then for convergence to w_{i+1}, $|g'(w_{i+1}^k)| < 1$ is required.

E.g. 2-step AB/AM p/c

$$|g'(w_{i+1})| = \Big| \frac{5h}{12} \underbrace{\frac{\partial f}{\partial y}\Big|_{w_{i+1}}}_{J} \Big| < 1$$

- Note that problems can occur when J is large. It requires small "h"

- We could also use other types of fixed pt iteration

E.g. Newton's method.

Rewrite $\underline{g(w_{i+1}) = w_{i+1}}$ as $F(w_{i+1}) = 0$.

$$F(w_{i+1}) = w_{i+1} - g(w_{i+1}) = 0.$$

In the case of 2 step AB/AM P/C, we get

$$F(w_{i+1}) = w_{i+1} - \left[w_i + \frac{h}{12}\left[5f_{i+1} + 8f_i - f_{i-1}\right]\right] = 0$$

Let $Z = w_{i+1}$

then the newtons method for $F(Z)$ will be

$$Z^{k+1} = Z^k - \frac{F(Z^k)}{F'(Z^k)}$$

$$w_{i+1}^{k+1} = w_{i+1}^k - \frac{w_{i+1}^k - \left(w_i + \frac{h}{12}\left[5f(w_{i+1}^k, t_{i+1}) + 8f_i - f_{i-1}\right)\right.}{1 - \frac{5h}{12}\left.\frac{\partial f}{\partial y}\right|_{w_{i+1}^k}}$$

for $k = 0, 1, \ldots$

Remarks:

- Convergence will be faster but requeres a good guess of $\tilde{w}_{i+1}$ (Prediction)
- In many problems, improving the corrected value is not worth it, much better way is to reduce the step size. [Reason: as before iteration converges to w_{i+1}

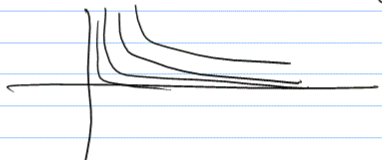
but not to y_{i+1}]

- For stiff equations (large J) there are some benifits in iteration.

- Eg:

$$y' = -ky \Rightarrow y = y_0 e^{-kt}$$

$$\frac{df}{dy} = -k, \quad J \text{ is large for large "} k \text{"}$$



Sytem of ODEs

- m first order ODE's in m variables and t

$$\frac{dY}{dt} = F(Y, t) \quad \text{where}$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \quad \text{and} \quad F = \begin{bmatrix} f_1(y_1, y_2, \cdots y_m, t) \\ \vdots \\ f_m(y_1, y_2, \cdots y_m, t) \end{bmatrix}$$

with initial Condition

$$Y(0) = \begin{bmatrix} y_1(0) \\ \vdots \\ y_m(0) \end{bmatrix}$$

- need to evaluate $Y(t)$
- explicit method is easy to implement but may have restriction on step size.
  - put y_i in a loop and do what we did for y.

- Implicit implies the use of predictor/corrector
  - must predict all values of $\{\tilde{y}_i\}$ before start Correction step.

E.g:

$$y_1' = y_2 y_3 - y_1 \qquad y_1(0) = \alpha$$

$$y_2' = y_1 y_3 - y_2 \qquad y_2(0) = \beta$$

$$y_3' = y_1 y_2 - y_3 \qquad y_3(0) = \gamma$$

Step 1:

Predict y_i^p ← Prediction at time t_i

Step 2:

Correct y_i^c ← Correction at time t_i using y_i^p

Ex: use 2-step AB/AM p/c method with single correction.

Remarks:

- Predicition can be done in any order but all must be predicted before correction.
- Correction can also be done in any order.

Higher order ODES

Eg. $y'' = f(y', y, t)$ with $y'(0) = \bar{y}_0$ and $y(0) = y_0$

Idea:

write a system of first order ODES, then

Solve with the methods that we have studied before.

In general

$$y^{m}(t) = f\left(t, y, y^{2}, y^{3}, \dots, y^{m-1}\right)$$

with initial conditions

$$y_0, \; y_0^{'}, \; y_0^{''}, \; \dots \; y_0^{m-1}$$

Let $Z_1(t) = y(t)$

$$Z_2(t) = y'(t) = Z_1'(t)$$

$$Z_3(t) = y''(t) = Z_2'(t)$$

$$\vdots$$

$$Z_m(t) = y^{m-1}(t) = \underline{Z_{m-1}'(t)}$$

$\Rightarrow$ initial conditions:

$$z_1(0) = y(0) = y_0$$

$$z_2(0) = y'(0) = y_0'$$

$$\vdots$$

$$z_m(0) = y^{m-1}(0) = y_0^{m-1}$$

Example:

Find $y(t)$, where $y''(t) + 4y'(t) + 5y(t) = 0$,

$y(0) = 3$, $y'(0) = 5$

Let $z_1(t) = y(t)$

$z_2(t) = y'(t) = z_1'(t)$

$\Rightarrow z_2'(t) = y''(t)$

$= -4y'(t) - 5y(t) = f(t, y')$

with $z_1(0) = y(0) = 3$

$z_2(0) = y'(0) = 5$

Using Euler:

$$z_1^{i+1} = z_1^i + h f_1^i$$
$$= z_1^i + h z_2^i$$
$$z_2^{i+1} = z_2^i + h f_2^i$$
$$= z_2^i + h(-4z_2^i - 5z_1^i)$$

Euler:
$y^{i+1} = y^i + h f^i$

$y' = f$

$z_1' = y_1' = f$

$z_2' = y_2' = f$

$$\Rightarrow \boxed{\begin{aligned} z_1^{i+1} &= z_1^i + h z_2^i \\ z_2^{i+1} &= z_2^i - h(4z_2^i + 5z_1^i) \end{aligned}}$$

for $i = 1, 2, \ldots$

$i = 1$

$$z_1^2 = z_1^1 + h z_2^1$$
$$z_2^2 = z_2^1 - h(4z_2^1 + 5z_1^1)$$

where $z_1^1 = 3$, $z_2^1 = 5$

Stability:

Euler's method:

$$\varepsilon_{i+1} = \underbrace{(1 + hJ_i)}_{\lambda \text{ — amplification factor.}} \varepsilon_i + \frac{h^2}{2} f'(\xi_i)$$

— error has both the truncation component and the propogation component.

Three facts about ODE Solution:

(i) Consistancy

Does the difference equation approaches the differential equation as $h \to 0$.

ie $\lim_{h \to 0} L.T.E = 0$?

Ans: all methods are based on Taylors expansion, hence this will be case always.

(ii) Stability :

Does errors, once introduced, remain bounded or do they grow exponentially?

- When the absolute value of the amplification factor is less than or equal to one, the error remains bounded, ie when $|\lambda| \leq 1$

(iii) Convergence

Does the numerical solution approach the exact solution as $h \to 0$

ie $\underset{\substack{\uparrow \\ \text{numerical sol}}}{u_h} \longrightarrow \underset{\substack{\uparrow \\ \text{exact}}}{u}$ as $h \to 0$

Equivalance Theorem:

A necessary and sufficient condition for

Convergence is the Consistancy and stability.

Note that the stability does not guarantee accuracy.

Stability of various methods:

Stability depends on

- step size
- ODE solution form (J)
- numerical method (LTE)

Backward Euler: (Implicit)

$$w_{i+1} = w_i + h f_{i+1}$$

It comes from the expansion $I(t) = \int_{t_i}^{t_{i+1}} f \, dt$

around t_{i+1} and evaluate at t_i

$$\boxed{e_{i+1} = y_{i+1} - w_{i+1}}$$

Error equation

$$y_{i+1} = y_i + h\, f(y_{i+1}, t_{i+1})$$

$$w_{i+1} = w_i + h\, \tilde{f}(w_{i+1}, t_{i+1})$$

$$e_{i+1} = e_i + h\left[f - \tilde{f}\right] \quad + \text{ error terms}$$

$$= e_i + h\left[\frac{f - \tilde{f}}{y_{i+1} - w_{i+1}}\right](y_{i+1} - w_{i+1})$$

$$= e_i + h J e_{i+1}$$

$$\Rightarrow \quad e_{i+1} = \underbrace{\left[1 - h J_{i+1}\right]^{-1}}_{\lambda} e_i$$

$$\frac{e_{i+1}}{e_i} = \frac{1}{1 - hJ} =$$

$$J < 0$$
$$-|J| < 0$$

Assume $J < 0$

$$\Rightarrow \quad \frac{e_{i+1}}{e_i} = \frac{1}{1 + h|J|} < 1$$

Given analytically stable ODE ie $J < 0$

$\Rightarrow$ the implicit Euler is unconditionally stable.

Exercise:

Check the stability condition of Explicit Euler scheme. (option (i) (or) option (ii))?

options: (i) unconditionally stable (or)

(ii) Conditionally stable (i.e stability depends on "h")

Stability of Trapezoidal scheme

$$y_{i+1} = y_i + \frac{h}{2}\left[f_i + f_{i+1}\right] + O(h^3)$$

$$w_{i+1} = w_i + \frac{h}{2}\left[\tilde{f}_i + \tilde{f}_{i+1}\right]$$

where $f_i = f(y_i, t_i)$ & $\tilde{f}_i = f(w_i, t_i)$

$\Rightarrow$ $e_{i+1} = e_i + \cdots$

$$e_{i+1} = \left(\frac{1 + \frac{h}{2} J_i}{1 - \frac{h}{2} J_{i+1}} \right) e_i$$

For $J < 0$ (analytically stable)

$$\frac{e_{i+1}}{e_i} = \frac{1 - \frac{h}{2}|J_i|}{1 + \frac{h}{2}|J_{i+1}|} < 1 \quad \text{for any } h > 0.$$

$\Rightarrow$ unconditionally stable!

Remarks:

- Backward Euler (implicit) is highly stable but may not be accurate as it is of only first order.
- implicit schemes are generally more stable
- Recall Trapezoidal rule is also an implicit scheme
- order of LTE in Trapezoidal $O(h^3)$ is even better than the implicit Euler $O(h^2)$

but for stiff problems amplification factor tends to one

Ex: Stiff Problem

$$y' = -ky \quad , \quad y(0) = y_0$$

Energy equation

$$\frac{1}{pe} \Delta T + b \cdot \nabla T = f$$

$$pe << 1$$

Optimization

- optimization is the term used for minimizing or maximizing a function.

- It is enough to consider the problem of minimization only, since the maximization of a function $F(x)$ can be obtained by simply minimizing "$-F(X)$".

- The function $F(x)$ is called the merit.

function or objective function.

— The Component "x" is called as a design variable.

— F(x) may have several local minima, the initial choice of x determines which of the minimum will be computed.

— Finding global minima is not guarented

— Types of optimization

(i) "Constrained optimization" : design variable is subjected to restrictions or constraints

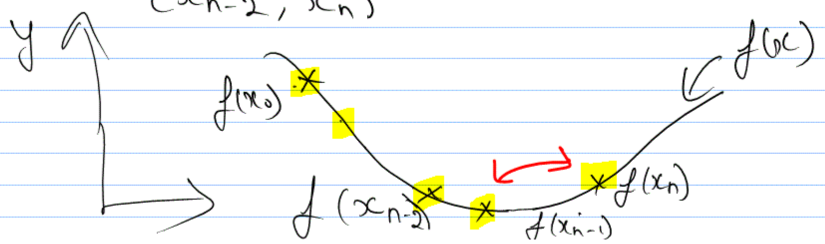
Eg: $\min f(x)$, where $x > 0$.

— "UnConstrained optimization": No restrictions are placed on the design variable.

Bracketing:

— minimum point must be bracketed before starting any minimization algorithm:

- start with an initial value x_0 and move downhill computing the function at $x_1, x_2, \ldots$
- stop at x_n, where $f(x_n)$ increases for the first time
- minimum point is bracketed in the interval (x_{n-2}, x_n)



Remarks:

1. The step size $h_i = x_{i+1} - x_i$ need not be a constant.
2. more efficient scheme is to increase h_i with every step
3. We could use $h_{i+1} = c\,h_i$, $c > 1$
4. Note that a local minimum might be missed for large value of "c".

Golden Section Search:

- It is a counterpart of the bisection method used to find roots of equations

  Recall: the root is bracketed in (a, b), then evaluates the function at $\left(x = \frac{a+b}{2}\right)$ and obtain a small interval eithter (a, x) or (x, b). This process continuous untill the interval is acceptably small.

- A minimum, by contrast, is bracketed with triplet $a < c < b$ (or) $b < c < a$ such that the function $f(c)$ is less than $f(b)$ and $f(a)$

Algorithm:

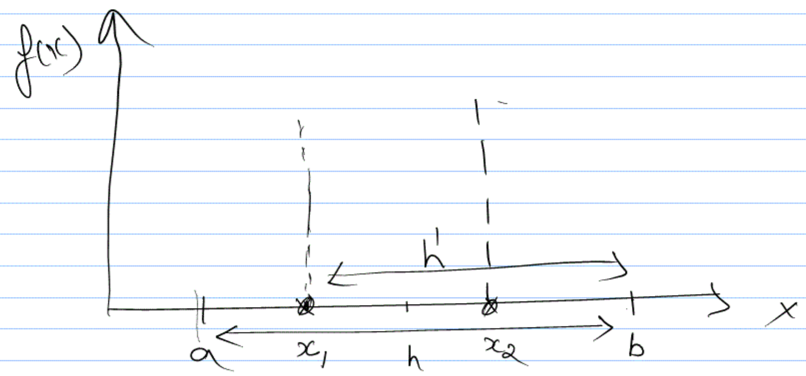
Suppose that the $\min f(x)$ has been bracketed in the interval (a, b) of length 'h', ie

(i) $h = b - a$

(ii) Let $x_1 = b - Rh$ and $x_2 = a + Rh$

with $R = 0.618033989$ (Golden ratio)

↑
Ratio of sides of
Golden rectangle



(iii) If $f(x_1) > f(x_2)$

$\Rightarrow$ min lies in (x_1, b)

Set $a = x_1$, $x_1 = x_2$ with

$h' = Rh$, $x_2 = a + Rh'$

Else:

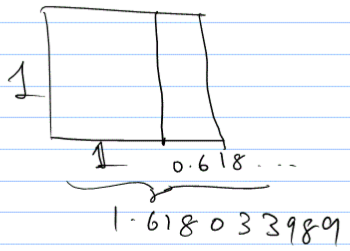
min lies in (a, x_2)

Set $b = x_2$, $x_2 = x_1$

where $h' = Rh$, $x_1 = b - Rh'$

Remark:

$R = 0.618033989$ is the golden ratio, where R is the ratio of a "golden rectangle" whose side lengths are in the ratio $1 : \frac{1+\sqrt{5}}{2}$



Multidimensional optimization:

The most commonly used methods are

(i) Downhill simplex method by Nelder and Mead

(ii) Powell's method

Downhill simplex method:

- Finding minimum of a function with more than one independent variable.

- It requires only function evaluations but not their derivatives.
- It is commonly used nonlinear optimization technique
- It has geometrical naturalness based on Simplex:
- An N-simplex is a geometrical figure of N+1 vertices (points) and all their interconnecting line segments, polygonal faces, etc..

Now three possibilities arise

(i) $f(x_1) \leq f(x_r) \leq f(x_n)$

reject x_{n+1} and set $x_{n+1} = x_r$

GoTo 2

ie x_r is better than the second worst but not better than the best

(ii) $f(x_r) < f(x_1)$

[x_r is better than the best]

Remarks:

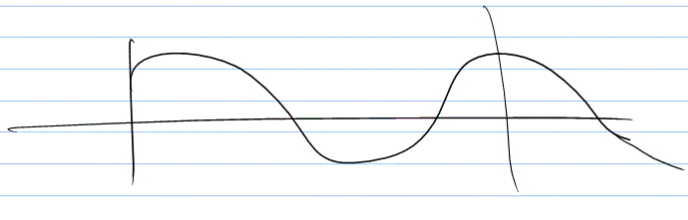
- For a quadratic function of n-independent variables, it is possible to construct "n" mutually conjugate direction.
- it takes precisely n-line minimizations along these directions to reach the minimum point.
- if F(x) in not a quadratic, it can be treated as a local approximation of the merit function obtained by truncating Taylor series expansion

Fourier Series:

Periodic functions:

A function $x(t)$ is periodic over time "T"

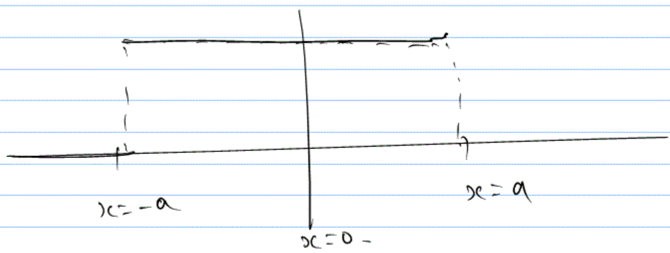
if
$$x(t+nT) = x(t)$$



$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega)\, e^{i\omega t}\, d\omega .$$

Eg:

$$f(t) = \begin{cases} 1 & |x| < a \\ 0 & \text{elsewhere} \end{cases}$$



Errors in DFT:

- DFT is only an approximation, since it provides only for a finite set of frequencies.
- It induce two types of error: (i) "aliasing" (ii) "leakage"

Alaising:

If the initial samples are not sufficiently closely spaced to represent high-

Basic idea in FFT

— Based on the fact that the DFT involves lot of redunant calculations.

Idea:

Rewrite
$$\hat{f}_n = \sum_{k=0}^{N-1} f_k \, w^{nk}$$

as summation over two summations, one for k-even and another for k-odd.

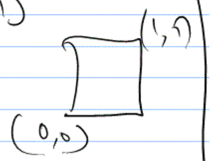
ε – diffusion co-eff

b – Convective velocity

<u>2D Analog:</u>

$$-\varepsilon \Delta u + b \cdot \nabla u = 0 \quad \text{in } (0,1)^2$$

+ B.Cs



<u>Solving by FD:</u>