NUMERICAL METHODS

CSE 2103

[Warning: this is not a complete package of the course.]

rmShoeb
CSE_16

Errors and their computations

The aim of numerical analysis is to provide efficient methods for obtaining numerical answers to such problems.

Approximate numbers: The numbers that represents a certain number of accuracy. Pi is an approximate number.

Rounding Off: In numerical computations, we come across some numbers which have large number of digits. It is necessary to oust them to a usable number of significant digits. This process is called rounding off. Ex.: 1.6583 to 1.658

Rounding means making a number simpler but keeping its value close to what it was. The result is less accurate, but easier to use. Example: 73 rounded to the nearest ten is 70, because 73 is closer to 70 than to 80. [mathisfun.com]

Numerical Error: In numerical calculations, some type of errors are encountered. These are called numerical errors.

Rounding off Error: Round-off errors originate from the fact that computers retain only a fixed number of significant figures during a calculation. Numbers such as π, e, or $\sqrt{7}$ cannot be expressed by a fixed number of significant figures. Therefore, they cannot be represented exactly by the computer. In addition, because computers use a base-2 representation, they cannot precisely represent certain exact base-10 numbers. The discrepancy introduced by this omission of significant figures is called *round-off error.*

A round-off error, also called rounding error, is the difference between the calculated approximation of a number and its exact mathematical value due to rounding. This is a form of quantization error. [Wikipedia]

Truncation Error: Error due to finite representation after inherently infinite process is called truncation error.

Or,

Truncation errors are those that result from using an approximation in place of an exact mathematical procedure.

In numerical analysis and scientific computing, truncation error is the error made by truncating an infinite sum and approximating it by a finite sum. For instance, if we approximate the sine function by the first two non-zero term of its Taylor series, as in for small, the resulting error is a truncation error. [Wikipedia]

What does the word truncation mean?

To shorten (a number) by dropping one or more digits after the decimal point. [thefreedictionary.com]

Absolute Error: It is the numerical difference between the true value of a quantity and its approximate value. If x is the true value and x_1 is the approximate value, then,

$$E_A = |x - x_1| = \delta x$$

Relative Error: It is defined as the ratio of the difference between the true value and approximate value and divided by its true value.

$$E_R = \frac{|x - x_1|}{x} = \frac{E_A}{x}$$

Percentage Error:

$$E_P = \frac{|x - x_1|}{x} \times 100 = 100E_R\%$$

$Example: An\ approximation\ value\ of\ \pi\ is\ {}^{22}/_{7} = 3.1428571\ and\ the\ real\ value\ is\ 3.1415926, then,$

$Absolute\ error = |3.1415926 - 3.14285714| = 0.0012645$

$Relative\ error = 0.0012645/3.1415926 = 4.025028 \times 10^{-4}$

$Percentage\ error = 4.025028 \times 10^{-4} \times 100 = 0.0402502$

Solution to Algebraic and Transcendental Equation

For Linear Algebraic Equations

1. Bisection Method
2. False position method
3. Newton-Raphson Method
4. Iteration method
5. Generalized Newton-Raphson Method
6. Ramanujan's Method
7. Secant Method

Bisection Method

The *bisection method,* which is alternatively called binary chopping, interval halving, or Bolzano's method, is one type of incremental search method in which the interval is always divided in half. If a function changes sign over an interval, the function value at the midpoint is evaluated. The location of the root is then determined as lying at the midpoint of the subinterval within which the sign change occurs. The process is repeated to obtain refined estimates.

$If\ f(x)\ is\ a\ continuous\ function\ between\ a\ and\ b, and\ f(a) and\ f(b)\ are\ of\ opposite\ signs,$
$then\ there\ exists\ at\ least\ one\ root\ between\ a\ and\ b\ and\ the\ approximate\ value\ will\ be$

$$x_0 = \frac{a+b}{2}$$

$If\ f(x_0) = 0, then\ x_0\ is\ the\ solution.$
$Else\ if\ f(x_0) > 0, then\ we\ have\ to\ update\ the\ value\ for\ which\ f(x_0) > 0.$
$Else\ if\ f(x_0) < 0, then\ we\ have\ to\ update\ the\ value\ for\ which\ f(x_0) < 0.$

Example: Find a real root of the equation:

$$x^3 - x - 1 = 0$$

Solution:
$Let, f(x) = x^3 - x - 1$
$a = 1 \qquad b = 2$
$So, f(a) = -1\ and\ f(b) = 5$
Here, f(b) is positive and f(a) is negative.

Iteration No.	a	b	$x_0 = \frac{a+b}{2}$	$f(x_0)$
1	1	2	1.5	0.875
2	1	1.5	1.25	-0.296875
3	1.25	1.5	1.375	0.224609375
4	1.25	1.375	1.3125	-0.05151367188
5	1.3125	1.375	1.34375	0.08261108398
6	1.3125	1.34375	1.328125	0.01457595825
7	1.3125	1.328125	1.3203125	-0.01871061325
8	1.3203125	1.328125	1.324218	-0.00212
9	1.324218	1.328125	1.326171	0.00624
10	1.324216	1.326171	1.32519	0.002026
11	1.324216	1.32519	1.324704	0.00006
12	1.324703	1.32519	1.3249465	0.000974
13	1.324703	1.324949	1.324824	0.000455
14	1.324703	1.324824	1.3247635	0.000194

15	1.324703	1.3247635	1.3247633	0.0000652

Since $f(x_{15}) < 0.0001$, which is very small, the root is 1.3247.

It should be noted that this method always succeeds. If there are more roots than one in the interval, this method finds one of the roots.

False Position Method

This is the oldest method for finding the real root of a non-linear equation and closely resembles the bisection method. It is also known as *regula falsi* or the *method of chords*.
In this method, we choose two points a and b such that f(a) and f(b) are opposite signs. Hence the approximate solution will be,

$$x_0 = \frac{af(b) - bf(a)}{f(b) - f(a)}$$

which is the first approximation to the root of f(x)=0. If now f(x_0) and f(a) are of opposite signs, then the root lies between a and x_0 and we replace b by x_0 and obtain the next iteration. The process is repeated till the root is obtained to the desired accuracy.

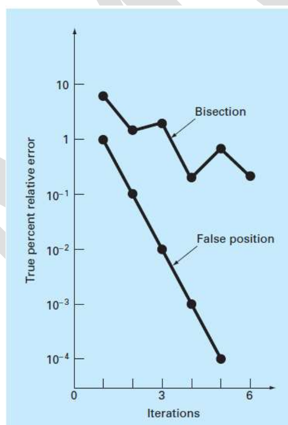


Figure: Comparison between Bisection and False Position method

Example: Find a real root of the equation

$$x^3 - 2x - 5 = 0$$

Solution:
$Let, f(x) = x^3 - 2x - 5$
$a = 2 \qquad b = 3$
$Here, f(a) = -1 \; and \; f(b) = 16$
Hence,

Iteration No.	a	b	$x = \frac{af(b) - bf(a)}{f(b) - f(a)}$	f(x)
1	2	3	2.058823529	-0.3908
2	2.058823529	3	2.08126366	-0.1472
3	2.08126366	3	2.089638211	-0.0547
4	2.089638211	3	2.092739575	-0.0202
5	2.092739575	3	2.093883709	-0.0075
6	2.093883709	3	2.094305451	-0.0027
7	2.094305451	3	2.094460845	-0.0010

Since $|x_7 - x_6| < 0.001$, which is very small, the root is 2.094.

Bisection method and False Position Method are also called **Bracketing Methods**.
[Bracketing method: These methods are based on making two initial guesses that bracket the root, that is, are on either side of the root.]

Iteration Method

$Let, f(x) = 0, \quad x = \varphi(x)\ and\ x_0 = initial\ guess.$
$Then\ x_n = \varphi(x_{n-1})\ \ and\ x_n\ will\ be\ the\ root\ of\ the\ equation\ if\ |x_n - x_{n-1}|\ is\ sufficiently\ small.$
$[If\ we\ can\ find\ more\ than\ one\ \varphi(x), then\ we\ will\ select\ the\ \varphi(x)\ for\ which\ |\varphi'(x)| < 1].$
$If\ there\ are\ more\ than\ one\ \varphi(x)\ for\ which\ \ |\varphi'(x)| < 1, then\ we\ can\ select\ any\ one\ of\ them.]$

Example: Find a real root of the equation,

$$x^3 = 1 - x^2$$

On the interval [0,1] with an accuracy of 10^{-4}.

Solution:
Given that,
$x^3 = 1 - x^2$
$\therefore x = \dfrac{1}{\sqrt{x+1}} = \varphi(x)$
$Let, x_0 = 0.5$

Iteration no.	x_{n-1}	$x_n = \varphi(x)$	$\lvert x_n - x_{n-1}\rvert$
1	0.5	0.816497	0.316497
2	0.816497	0.741964	0.074533
3	0.741964	0.757671	0.015707
4	0.757671	0.754278	0.003393
5	0.754278	0.755007	0.000729
6	0.755007	0.75485	0.000757
7	0.75485	0.754884	0.000034

Hence, the root is 0.754884.

[Shortcut technique: যখন function এ logarithmic function, trigonometric function অথবা exponential function (like: e^x) থাকবে, তখন initial guess 3.2 ধরব, অন্যান্য ক্ষেত্রে 0.5 ধরব।

Newton-Raphson Method

$Let, x_0\ be\ an\ approximate\ root\ of\ f(x) = 0\ and\ x_1 = x_0 + h\ be\ the\ correct\ root\ so\ that,$
$f(x_1) = 0.$
$Expanding\ f(x_0 + h)$ by Taylor's series, we obtain,

$$f(x_0) + hf'(x_0) + \frac{h^2}{2!}f''(x_0) + \cdots = 0$$

Neglecting the second and higher order derivatives, we have,

$$f(x_0) + hf'(x_0) = 0$$

Which gives,

$$h = -\frac{f(x_0)}{f'(x_0)}$$

Hence,

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

For successive iterations,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Example: Find a real root for the equation,

$$x^3 - 2x - 5 = 0$$

Solution:

$f(x) = x^3 - 2x - 5$
$f'(x) = 3x^2 - 2$
$Let, x_0 = 2$

Iteration no.	x_n	$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
0	2	2.1
1	2.1	2.094568
2	2.094568	2.094551
3	2.094551	2.094551

Since, $|x_{n-1} - x_n| < 0.0001$, hence, the root is 2.094551.

Generalized Newton-Raphson Method [Edit]

$If\ x\ is\ a\ root\ of\ f(x) = 0\ with\ multiplicity\ p, then\ the\ iteration\ formula\ is\ given\ by,$

$$x_{n+1} = x_n - p\frac{f(x_n)}{f'(x_n)}$$

Ramanujan's Method

This method is used to determine the smallest root of the equation,

$$f(x) = 0$$

Where, f(x) is of the form

$$f(x) = 1 - (a_1x + a_2x^2 + a_3x^3 + \cdots)$$

Let,

$[1 - (a_1x + a_2x^2 + a_3x^3 + \cdots)]^{-1} = b_1 + b_2x + b_3x^2 + \cdots$
$\Rightarrow 1 + (a_1x + a_2x^2 + a_3x^3 + \cdots) + (a_1x + a_2x^2 + a_3x^3 + \cdots)^2 + \cdots = b_1 + b_2x + b_3x^2 + \cdots$

Equating the co-efficient on both sides, we obtain,

$b_1 = 1$
$b_2 = a_1b_1$
$b_3 = a_1b_2 + a_2b_1$
.. ..
.. ..
$b_n = a_1b_{n-1} + a_2b_{n-2} + \cdots + a_{n-1}b_1$

The ratios $\frac{b_{n-1}}{b_n}$, called the convergent, approach, in the limit, the smallest root of f(x) =0.

Example: Find the smallest root of the equation

$$x^3 - 9x^2 + 26x - 24 = 0$$

Solution:

Given that,

$$x^3 - 9x^2 + 26x - 24 = 0$$

$$\therefore f(x) = 1 - \left(\frac{13}{12}x - \frac{3}{8}x^2 + \frac{1}{24}x^3\right)$$

$So, \quad a_1 = \frac{13}{12}, \quad a_2 = \frac{-3}{8}, \quad a_3 = \frac{1}{24}$

Hence,

n	b_n	$\frac{b_{n-1}}{b_n}$	$\left\|\frac{b_{n-2}}{b_{n-1}} - \frac{b_{n-1}}{b_n}\right\|$
1	1	-	-
2	1.083333	0.9230769231	-
3	0.7986111111	1.3565217391	0.433448079
4	0.5005787037	1.5953757225	0.2388539915
5	0.2879533179	1.7384022777	0.1430265552

The process will go on until $\left|\frac{b_{n-2}}{b_{n-1}} - \frac{b_{n-1}}{b_n}\right|$ is sufficiently small. When we find a really small value for this, like less than 0.0001, then the corresponding $\frac{b_{n-1}}{b_n}$ is the root.
In this case, the root is 1.99998 ≈ 2.

Secant Method

The Newton-Raphson method requires the evaluation derivatives of the function and this is not always not possible, particularly in the case of functions arising in practical problems. In the Secant method, the derivative at x_i is approximated by the formula,

$$f'(x_i) \approx \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}$$

Hence, the Newton-Raphson formula becomes

$$x_{i+1} = x_i - \frac{f(x_i)}{\frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}} = x_i - \frac{x_i f(x_i) - x_{i-1} f(x_i)}{f(x_i) - f(x_{i-1})}$$

$$= \frac{x_i f(x_i) - x_i f(x_{i-1}) - x_i f(x_i) + x_{i-1} f(x_i)}{f(x_i) - f(x_{i-1})}$$

$$\therefore x_{i+1} = \frac{x_{i-1} f(x_i) - x_i f(x_{i-1})}{f(x_i) - f(x_{i-1})}$$

Oh, and I forgot to mention that this method requires two initial approximation to the root. So, chill :p

Example: *Find a real root of the equation* $x^3 - 2x - 5 = 0$.
Sol^n:
Let the two initial approximation be given by $x_0 = 2$ *and* $x_1 = 3$
Then,

i	x_{i-1}	x_i	$x_{i+1} = \frac{x_{i-1} f(x_i) - x_i f(x_{i-1})}{f(x_i) - f(x_{i-1})}$
1	2	3	2.058824
2	3	2.058824	2.081264
3	2.058824	2.081264	2.094824
4	2.081264	2.094824	2.094549
5	2.094824	2.094549	2.094551

Since, $|x_6 - x_5| < 0.0001$, *which is very small, hence, the root of the given equation is* 2.094551.

Iteration Method, Newton-Raphson Method, Generalized Newton-Raphson Method and Secant Method are also called **Open Method**.
[Open method: These methods do not need boundary values. Converges quickly and may not be able to a root even if there is any. You may think that the Secant method is a bracketing

method because it needs two initial guess. But it is not since the initial values do not bracket the root, not always.]

For Non-Linear Algebraic Equations

1. Iteration Method
2. Newton-Raphson Method

Iteration Method

Let us assume that,

$$f(x,y) = 0 \text{ and } g(x,y) = 0$$
$$x = F(x,y) \text{ and } y = G(x,y)$$

Where the function F and G satisfy the condition

$$\left|\frac{\partial F}{\partial x}\right| + \left|\frac{\partial F}{\partial y}\right| < 1 \quad and \quad \left|\frac{\partial G}{\partial x}\right| + \left|\frac{\partial G}{\partial y}\right| < 1$$

If (x_0, y_0) is an initial approximation, then,

$$x_1 = F(x_0, y_0), \qquad y_1 = G(x_0, y_0)$$
$$x_2 = F(x_1, y_1), \qquad y_2 = G(x_1, y_1)$$
$$\vdots \qquad\qquad \vdots$$
$$x_{n+1} = F(x_n, y_n), \qquad y_{n+1} = G(x_n, y_n)$$

(x_{n+1}, y_{n+1}) *is the root if* $|x_{n+1} - x_n|$ *and* $|y_{n+1} - y_n|$ *is sufficiently small.*

Example: *Find a real root of the equations*

$$y^2 - 5y + 4 = 0$$
$$3yx^2 - 10x + 7 = 0$$

Sol^n:

Given that,

$$y^2 - 5y + 4 = 0 \Rightarrow y = \frac{1}{5}(y^2 + 4) = G(x,y) \quad and$$

$$3yx^2 - 10x + 7 = 0 \Rightarrow x = \frac{1}{10}(3yx^2 + 7) = F(x,y)$$

$$\therefore \frac{\partial F}{\partial x} = \frac{3xy}{5}, \quad \frac{\partial F}{\partial y} = \frac{3x^2}{10}, \quad \frac{\partial G}{\partial x} = 0, \quad \frac{\partial G}{\partial y} = \frac{2y}{5}$$

$Let, x_0 = 0.5 \text{ and } y_0 = 0.5$

$$\left|\frac{\partial F}{\partial x}\right| + \left|\frac{\partial F}{\partial y}\right| = \left|\frac{3xy}{5}\right| + \left|\frac{3x^2}{10}\right| = 0.15 + 0.075 < 1$$

$$and \left|\frac{\partial G}{\partial x}\right| + \left|\frac{\partial G}{\partial y}\right| = |0| + \left|\frac{2y}{5}\right| = 0.2 < 1$$

Hence,

n	x_n	y_n	$x_{n+1} = F(x,y)$	$x_{n+1} = G(x,y)$
0	0.5	0.5	0.7375	0.85
1	0.7375	0.85	0.8387	0.9445
2	0.8387	0.9445	0.8993	0.9784
3	0.8993	0.9784	0.9374	0.9914
4	0.9374	0.9914	0.9613	0.9966
5	0.9613	0.9966	0.9763	0.9986
6	0.9763	0.9986	0.9855	0.9994

And the root will be (1,1).

Newton-Raphson Method

Let us assume that,

$$f(x,y)=0 \text{ and } g(x,y)=0$$

Let (x_0,y_0) *be an initial approximation to the root. If* (x_0+h, y_0+k) *is the actual root of the system, then we must have,*

$$f(x_0+h, y_0+k)=0 \text{ and } g(x_0+h, y_0+k)=0$$

Assuming that f and g are sufficiently differentiable, expanding both functions using Taylor's series,

$$f_0+h\frac{\partial f}{\partial x_0}+k\frac{\partial f}{\partial y_0}+\cdots=0$$
$$g_0+h\frac{\partial g}{\partial x_0}+k\frac{\partial g}{\partial y_0}+\cdots=0$$

Neglecting second and higher order terms,

$$h\frac{\partial f}{\partial x_0}+k\frac{\partial f}{\partial y_0}=-f_0$$
$$\text{and } h\frac{\partial g}{\partial x_0}+k\frac{\partial g}{\partial y_0}=-g_0$$

By Cramer's rule,

$$D=\begin{vmatrix}\frac{\partial f}{\partial x_0} & \frac{\partial f}{\partial y_0}\\ \frac{\partial g}{\partial x_0} & \frac{\partial g}{\partial y_0}\end{vmatrix}\neq 0$$

$$h=\frac{1}{D}\begin{vmatrix}-f_0 & \frac{\partial f}{\partial y_0}\\ -g_0 & \frac{\partial g}{\partial y_0}\end{vmatrix} \text{ and } k=\frac{1}{D}\begin{vmatrix}\frac{\partial f}{\partial x_0} & -f_0\\ \frac{\partial g}{\partial x_0} & -g_0\end{vmatrix}$$

The new approximations are therefore, $x_1=x_0+h$ and $y_1=y_0+k$

(x_{n+1},y_{n+1}) *is the root if* $|x_{n+1}-x_n|$ *and* $|y_{n+1}-y_n|$ *is sufficiently small.*

Example: Solve the equations,

$$y^2-5y+4=0$$
$$3yx^2-10x+7=0$$

Solution:

Let,

$f(x)=3yx^2-10x+7 \quad \text{and} \quad g(x)=y^2-5y+4$

$\therefore \frac{\partial f}{\partial x}=6xy-10, \quad \frac{\partial f}{\partial y}=3x^2$

$\frac{\partial g}{\partial x}=0, \quad \frac{\partial g}{\partial y}=2y-5$

Taking $x_0=0.5$ *and* $y_0=0.5$

$\therefore \frac{\partial f}{\partial x_0}=-8.5, \quad \frac{\partial f}{\partial y_0}=0.75, \quad f_0=2.375$

$\frac{\partial g}{\partial x_0}=0, \quad \frac{\partial g}{\partial y_0}=-4, \quad g_0=1.75$

$Hence,\quad D = \begin{vmatrix} -8.5 & 0.75 \\ 0 & -4 \end{vmatrix} = 34$

$Therefore,\quad h = \frac{1}{34}\begin{vmatrix} -2.375 & 0.75 \\ -1.75 & -4 \end{vmatrix} = 0.3180$

$and\quad k = \frac{1}{34}\begin{vmatrix} -8.5 & -2.375 \\ 0 & -1.75 \end{vmatrix} = 0.4375$

$It\ follows\ that, x_1 = x_0 + h = 0.5 + 0.3180 = 0.8180$

$and\quad y_1 = y_0 + k = 0.5 + 0.4375 = 0.9375$

The same process will go on……

Hence the root will be (1,1).

Interpolation

Given the set of tabulated values $(x_0, y_0), (x_1, y_1), \ldots\ldots\ldots, (x_n, y_n)$ satisfying the relation y=f(x), where the explicit nature of f(x) is not known. It is required to find a simpler function say φ(x) such that f(x) and φ(x) agree at the set of tabulated points. Such process is called ***Interpolation***. Here, if φ(x) is a polynomial, then it is called polynomial interpolation and φ(x) is called interpolating polynomial.

Finite Differences

Forward Differences:

If $y_0, y_1, y_2, \ldots\ldots\ldots, y_n$ are the tabular values of y and $y_1 - y_0, y_2 - y_1, \ldots\ldots, y_n - y_{n-1}$ *are the* differences of y. Denoting these differences by $\Delta y_0, \Delta y_1, \ldots\ldots, \Delta y_{n-1}$ respectively, we have,

$$\Delta y_0 = y_1 - y_0, \quad \Delta y_1 = y_2 - y_1, \quad \ldots\ldots, \quad \Delta y_{n-1} = y_n - y_{n-1}$$

$where\ \Delta\ is\ called\ the\ forward\ difference\ operator\ and\ \Delta y_0, \Delta y_1, \ldots\ are\ called\ first\ forward$ differences. The differences of the first forward differences are called second forward differences and are denoted by $\Delta^2 y_0, \Delta^2 y_1, \ldots\ldots$

Backward Differences:

$The\ differences\ y_1 - y_0, y_2 - y_1, \ldots\ldots, y_n - y_{n-1}\ are\ called\ first\ backward\ differences\ if\ they\ are\ denoted\ by\ \nabla y_1, \nabla y_2, \ldots\ldots, \nabla y_n\ respectively, so\ that,$

$$\nabla y_1 = y_1 - y_0,\ \ \nabla y_2 = y_2 - y_1, \ldots\ldots \nabla y_n = y_n - y_{n-1}$$

$where\ \nabla\ is\ called\ the\ backward\ difference\ operator.$

Central Differences

$The\ central\ difference\ operator\ \delta\ is\ defined\ by\ the\ relations$

$$y_1 - y_0 = \delta y_{1/2}, \quad y_2 - y_1 = \delta y_{3/2}, \ldots\ldots, \quad y_n - y_{n-1} = \delta y_{n-1/2}$$

Similarly, higher order differences can be defined.

Newton's formula for Interpolation

$Given\ the\ set\ of\ (n+1)\ valus, viz, (x_0, y_0), (x_1, y_1), (x_2, y_2) \ldots\ldots\ldots, (x_n, y_n), of\ \ x\ and\ y, it\ is\ required\ to\ find\ y_n(x), a\ polynomial\ of\ the\ n-th\ degree\ such\ that\ y\ and\ y_n(x)\ \ agree\ at\ the\ tabulated\ points.$

$Let\ the\ values\ of\ x\ be\ equidistant, i.e.\ let,$

$$x_i = x_0 + ih, \qquad i = 0,1,2, \ldots\ldots\ldots\ldots\ldots\ldots, n$$

Since $y_n(x)$ is a polynomial of the nth degree, it may be written as,

$$y_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) + \cdots \ldots\ldots\ldots\ldots\ldots$$
$$+ a_n(x - x_0)(x - x_1)(x - x_2) \ldots\ldots\ldots\ldots (x - x_{n-1})$$

Imposing now the condition that y and $y_n(x)$ should agree at the set of tabulated points, we obtain,

$$a_0 = y_0;\ a_1 = \frac{y_1 - y_0}{x_1 - x_0} = \frac{\Delta y_0}{h};\ \ a_2 = \frac{\Delta^2 y_0}{h^2 2!};\ a_3 = \frac{\Delta^3 y_0}{h^3 3!};\dots\dots;a_n = \frac{\Delta^n y_0}{h^n n!};$$

$Setting\ x = x_0 + ph\ and\ substituting\ for\ a_0, a_1, \dots\dots\dots, a_n\ gives,$

$$y_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots$$
$$+ \frac{p(p-1)(p-2)\dots(p-n+1)}{n!}\Delta^n y_0$$

Which is Newton's Forward Difference Interpolation formula and is useful for interpolation near the beginning of a set of tabular values.

Instead of assuming $y_n(x)$, if we assume it in the form,

$$y_n(x) = a_0 + a_1(x - x_n) + a_2(x - x_n)(x - x_{n-1}) + a_3(x - x_n)(x - x_{n-1})(x - x_{n-2}) +$$
$$\cdots\dots\dots\dots\dots.. + a_n(x - x_n)(x - x_{n-1})(x - x_{n-2})\dots\dots\dots\dots(x - x_1)$$

And then impose the condition that y and $y_n(x)$ should agree at the tabulated points $x_n, x_{n-1}, \dots\dots, x_2, x_1, x_0$, we obtain,

$$y_n(x) = y_n + p\nabla y_n + \frac{p(p+1)}{2!}\nabla^2 y_n + \cdots\dots\dots + \frac{p(p+1)\dots(p+n-1)}{n!}\nabla^n y_n;\ where, p = \frac{x - x_n}{h}.$$

This is Newton's Backward Difference Interpolation formula and it uses tabular values to the left of y_n. This formula is, therefore, useful for interpolation near the end of the tabular values.

Gauss' Central Difference Formulae

Gauss' forward formula

Let's consider the following difference table where the central ordinate is taken for the convenience as $y_0\ corresponding\ to\ x = x_0$.

The differences used in this formula lie on the line shown on the table. The formula is, therefore, of the form,

$$y_p = y_0 + G_1\Delta y_0 + G_2\Delta^2 y_{-1} + G_3\Delta^3 y_{-1} + G_4\Delta^4 y_{-2} +\ \dots\dots\dots (1)$$

$where\ G_1, G_2, \dots\dots have\ to\ be\ determined. The\ y_p\ on\ the\ left\ side\ can\ be\ expressed\ in\ the\ term\ of\ y_0, \Delta y_0\ and\ higher\ order\ differences\ of\ y_0, as\ follows:$

x	y	Δ	Δ^2	Δ^3	Δ^4	Δ^5	Δ^6
x_{-3}	y_{-3}						
		Δy_{-3}					
x_{-2}	y_{-2}		$\Delta^2 y_{-3}$				
		Δy_{-2}		$\Delta^3 y_{-3}$			
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$		$\Delta^4 y_{-3}$		
		Δy_{-1}		$\Delta^3 y_{-2}$		$\Delta^5 y_{-3}$	
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$		$\Delta^6 y_{-3}$
		Δy_0		$\Delta^3 y_{-1}$		$\Delta^5 y_{-2}$	
x_1	y_1		$\Delta^2 y_0$		$\Delta^4 y_{-1}$		
		Δy_1		$\Delta^3 y_0$			
x_2	y_2		$\Delta^2 y_1$				
		Δy_2					
x_3	y_3						

$$y_p = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 +\ \dots\dots\dots$$

(Where did this come from? Well, it's from Newton's forward difference interpolation.)

Similarly,

$$\Delta^2 y_{-1} = \Delta^2 y_0 - \Delta^3 y_0 + \Delta^4 y_0 - \Delta^5 y_0 +\ \dots\dots\dots$$
$$\Delta^3 y_{-1} = \Delta^3 y_0 - \Delta^4 y_0 + \Delta^5 y_0 - \Delta^6 y_0 +\ \dots\dots\dots$$
$$\Delta^4 y_{-2} = \Delta^4 y_0 - 2\Delta^5 y_0 + 3\Delta^6 y_0 - 4\Delta^7 y_0 +\ \dots\dots\dots$$

(And where did these come from? I have no ****ing idea.)

Hence, equation (1) gives the identity,

$$y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \frac{p(p-1)(p-2)(p-3)}{4!}\Delta^4 y_0 +\ \dots\dots =$$

$$y_0 + G_1\Delta y_0 + G_2(\Delta^2 y_0 - \Delta^3 y_0 + \Delta^4 y_0 - \Delta^5 y_0 + \ \dots) + G_3(\Delta^3 y_0 - \Delta^4 y_0 + \Delta^5 y_0 - \Delta^6 y_0 + \ \dots) +$$
$$G_4(\Delta^4 y_0 - 2\Delta^5 y_0 + 3\Delta^6 y_0 - 4\Delta^7 y_0 + \ \dots) + \ \dots\dots\dots$$

Equating the co-efficient from both sides of the equation, we obtain,

$$G_1 = p$$
$$G_2 = \frac{p(p-1)}{2!}$$
$$G_3 = \frac{(p+1)p(p-1)}{3!}$$
$$G_4 = \frac{(p+1)p(p-1)(p-2)}{4!}$$

Gauss' backward formula

x	y	Δ	Δ^2	Δ^3	Δ^4	Δ^5	Δ^6
$\vdots$	$\vdots$						
x_{-1}	y_{-1}						
		Δy_{-1}		$\Delta^3 y_{-2}$		$\Delta^5 y_{-3}$	
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$		$\Delta^6 y_{-3}$
		Δy_0		$\Delta^3 y_{-1}$		$\Delta^5 y_{-2}$	
x_1	y_1						
$\vdots$	$\vdots$						

Gauss' backward formula can therefore be assumed to be of the form,

$$y_p = y_0 + G_1'\Delta y_{-1} + G_2'\Delta^2 y_{-1} + G_3'\Delta^3 y_{-2} + G_4'\Delta^4 y_{-2} + \ \dots\dots\dots$$

Following the same procedure in Gauss' forward formula, we obtain,

$$G_1' = p$$
$$G_2' = \frac{p(p+1)}{2!}$$
$$G_3' = \frac{(p+1)p(p-1)}{3!}$$
$$G_4' = \frac{(p+2)(p+1)p(p-1)}{4!}$$

$$\left[Note: p = \frac{x - x_0}{h}, \qquad h = x_n - x_{n-1}\ \{for\ both\ forward\ and\ backward\ interpolation\}\right]$$

Stirling's Formula

Taking the mean of Gauss' forward and backward interpolation formula, we obtain,

$$y_p = y_0 + \frac{\Delta y_{-1} + \Delta y_0}{2} + \frac{p^2}{2}\Delta^2 y_{-1} + \frac{p(p^2-1)}{3!}\frac{\Delta^3 y_{-1} + \Delta^3 y_{-2}}{2} + \frac{p^2(p^2-1)}{4!}\Delta^4 y_{-2} + \ \dots\dots$$

This is the Stirling's formula.

Interpolation with unevenly spaced points

Lagrange's Interpolation Formula

Let $y(x)$ be continuous and differentiable $(n+1)$ times in the interval (a,b). Given the $(n+1)$ points $(x_0,y_0),(x_1,y_1),\dots\dots,(x_n,y_n)$ where the values of x need not necessarily be eq-ually spaced, we wish to find a polynomial of degree n, say $L_n(x)$. Let's say, the equation of a straight line passing through two points (x_0,y_0) and (x_1,y_1), then,

$$L_1(x) = \frac{x - x_1}{x_0 - x_1}y_0 + \frac{x - x_0}{x_1 - x_0}y_1$$

This is the Lagrange polynomial of degree one passing through two points. In a similar way, Lagrange polynomial of degree two passing through three points will be,

$$L_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}y_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}y_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}y_2$$

So, the general formula will be,

$$L_n(x) = a_0y_0 + a_1y_1 + \dots + a_ny_n$$

$$where, \qquad a_i = \frac{(x-x_0)(x-x_1)\dots\dots(x-x_{i-1})(x-x_{i+1})\dots\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots\dots(x_i-x_n)}$$

Least Squares Curve Fitting Procedures

Let, the set of data points be (x_1, y_1) *where,* $i = 1,2,\dots,n$ *and the curve given by* $Y = f(x)$ *be fitted to this data. At* $x = x_i$*, the experimented value on the ordinate is* y_i *and the cor − responding value on the fitting curve is* $f(x_i)$*. If* e_i *is the error of approximation at* $x = x_i$*, then we have,* $e_i = y_i - f(x_i)$

If we write,

$$S = [y_1 - f(x_1)]^2 + [y_2 - f(x_2)]^2 + \dots + [y_m - f(x_m)]^2$$
$$= e_1^2 + e_2^2 + \dots + e_m^2$$

Fitting a straight line

Let, $Y = a_0 + a_1 x$ *be the stright line to be fitted to the given data. Then we have,*

$$S = [y_1 - (a_0 + a_1 x_1)]^2 + [y_2 - (a_0 + a_1 x_2)]^2 + \dots + [y_m - (a_0 + a_1 x_m)]^2$$

for S to be minimum, we have,

$$\frac{\partial S}{\partial a_0} = 0 \qquad \frac{\partial S}{\partial a_1} = 0$$

$$\frac{\partial S}{\partial a_0} = -2[y_1 - (a_0 + a_1 x_1)] - 2[y_2 - (a_0 + a_1 x_2)] - \dots - 2[y_m - (a_0 + a_1 x_m)]$$

$$\therefore [y_1 - (a_0 + a_1 x_1)] + [y_2 - (a_0 + a_1 x_2)] + \dots + [y_m - (a_0 + a_1 x_m)] = 0$$

$$\Rightarrow y_1 + y_2 + \dots + y_m = m a_0 + a_1(x_1 + x_2 + \dots + x_m)$$

$$\Rightarrow \sum_{i=1}^{m} y_i = m a_0 + a_1 \sum_{i=1}^{m} x_i$$

$$\frac{\partial S}{\partial a_1} = -2x_1[y_1 - (a_0 + a_1 x_1)] - 2x_2[y_2 - (a_0 + a_1 x_2)] - \dots - 2x_m[y_m - (a_0 + a_1 x_m)]$$

$$\therefore x_1[y_1 - (a_0 + a_1 x_1)] + x_2[y_2 - (a_0 + a_1 x_2)] + \dots + x_m[y_m - (a_0 + a_1 x_m)] = 0$$

$$\Rightarrow x_1 y_1 + x_2 y_2 + \dots + x_m y_m = a_0(x_1 + x_2 + \dots + x_m) + a_1(x_1^2 + x_2^2 + \dots + x_m^2)$$

$$\Rightarrow \sum_{i=1}^{m} x_i y_i = a_0 \sum_{i=1}^{m} x_i + a_1 \sum_{i=1}^{m} x_i^2$$

Polynomial of nth Degree

Let the polynomial of nth degree,

$$Y = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

be fitted to the data points (x_i, y_i), $i = 1,2,\dots,m$*. Then, we have,*

$$S = [y_1 - (a_0 + a_1 x_1 + \dots + a_n x_1^n)]^2 + [y_2 - (a_0 + a_1 x_2 + \dots + a_n x_2^n)]^2 + \dots + [y_m - (a_0 + a_1 x_m + \dots + a_n x_m^n)]^2$$

Equating the first partial derivatives,

$$m a_0 + a_1 \sum_{i=1}^{m} x_i + a_2 \sum_{i=1}^{m} x_i^2 + \dots + a_n \sum_{i=1}^{m} x_i^n = \sum_{i=1}^{m} y_i$$

$$a_0 \sum_{i=1}^{m} x_i + a_1 \sum_{i=1}^{m} x_i^2 + a_2 \sum_{i=1}^{m} x_i^3 + \dots + a_n \sum_{i=1}^{m} x_i^{n+1} = \sum_{i=1}^{m} x_i y_i$$

$$\vdots$$

$$a_0 \sum_{i=1}^{m} x_i^n + a_1 \sum_{i=1}^{m} x_i^{n+1} + a_2 \sum_{i=1}^{m} x_i^{n+2} + \dots + a_n \sum_{i=1}^{m} x_i^{2n} = \sum_{i=1}^{m} x_i^n y_i$$

As we will be dealing with polynomial of 2nd degree here, the equations will be,

$$ma_0 + a_1 \sum_{i=1}^{m} x_i + a_2 \sum_{i=1}^{m} x_i^2 = \sum_{i=1}^{m} y_i$$

$$a_0 \sum_{i=1}^{m} x_i + a_1 \sum_{i=1}^{m} x_i^2 + a_2 \sum_{i=1}^{m} x_i^3 = \sum_{i=1}^{m} x_i y_i$$
$$a_0 \sum_{i=1}^{m} x_i^2 + a_1 \sum_{i=1}^{m} x_i^3 + a_2 \sum_{i=1}^{m} x_i^4 = \sum_{i=1}^{m} x_i^2 y_i$$

Exponential Function
Let the curve

$$y = ae^{bx}$$

be fitted to the given data. Taking logarithms of both sides, we get,

$$\log_e y = \log_e a + bx \log_e e$$
$$\Rightarrow \ln y = \ln a + bx$$

which can be written in the form,

$$Z = A + bx$$
$$where, \quad Z = \ln y, \quad A = \ln a$$

The problem therefore reduces to a least squares straight line through the given data.

Numerical Differentiation and Integration

Numerical Differentiation

Considering Newton's Forward difference formula-

$$y_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots, where, x = x_0 + ph$$

Then,

$$\frac{dy}{dx} = \frac{dy}{dp}.\frac{dp}{dx} = \frac{1}{h}\left(\Delta y_0 + \frac{2p-1}{2}\Delta^2 y_0 + \frac{3p^2-6p+2}{6}\Delta^3 y_0 + \cdots\right)$$

Differentiating again, we get,

$$\frac{d^2y}{dx^2} = \frac{1}{h^2}\left(\Delta^2 y_0 + \frac{6p-6}{6}\Delta^3 y_0 + \frac{12p^2-36p+22}{24}\Delta^4 y_0 + \cdots\right)$$

This formula can be used to compute the value of $\frac{dy}{dx}$ *and* $\frac{d^2y}{dx^2}$ for non-tabular values of x. For tabular values, we set,

$$x = x_0 \quad and\ hence, p = 0$$

Hence, the formula becomes,

$$\left[\frac{dy}{dx}\right]_{x=x_0} = \frac{1}{h}\left[\Delta y_0 - \frac{1}{2}\Delta^2 y_0 + \frac{1}{3}\Delta^3 y_0 - \frac{1}{4}\Delta^4 y_0 + \cdots\right] \ and$$
$$\left[\frac{d^2y}{dx^2}\right]_{x=x_0} = \frac{1}{h^2}\left[\Delta^2 y_0 - \Delta^3 y_0 + \frac{11}{12}\Delta^4 y_0 + \cdots\right]$$

Again, using Newton's backward difference formula,

$$\left[\frac{dy}{dx}\right]_{x=x_n} = \frac{1}{h}\left(\nabla y_n + \frac{1}{2}\nabla^2 y_n + \frac{1}{3}\nabla^3 y_n + \cdots\right)$$
$$\left[\frac{d^2y}{dx^2}\right]_{x=x_n} = \frac{1}{h^2}\left(\nabla^2 y_n + \nabla^3 y_n + \frac{11}{12}\nabla^4 y_n + \frac{5}{6}\nabla^5 y_n + \cdots\right)$$

Maximum and Minimum values of a tabulated function

It is known that the maximum and minimum values of a function can be found by equating the first derivative to zero and solving for the variable. The same procedure can be applied to determine the maxima and minima of a tabulated function.
Considering Newton's Forward difference formula-

$$y_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots, where, x = x_0 + ph$$

Differentiating this with respect to p, we get,

$$\frac{dy}{dx} = \frac{dy}{dp}.\frac{dp}{dx} = \frac{1}{h}\left(\Delta y_0 + \frac{2p-1}{2}\Delta^2 y_0 + \frac{3p^2-6p+2}{6}\Delta^3 y_0 + \cdots\right)$$

For maxima and minima $\frac{dy}{dp} = 0$. Hence, terminating the right hand side, for simplicity, after the third difference and equating it to zero, we obtain the quadratic for p,

$$c_0 + c_1 p + c_2 p^2 = 0$$

Where,

$$c_0 = \Delta y_0 - \frac{1}{2}\Delta^2 y_0 + \frac{1}{3}\Delta^3 y_0$$
$$c_1 = \Delta^2 y_0 - \Delta^3 y_0$$
$$c_2 = \frac{1}{2}\Delta^3 y_0$$

Numerical Integration

Given a set of data points $(x_0, y_0), (x_1, y_1), \dots\dots, (x_n, y_n)$ of a function $y = f(x)$, where f(x) is not known explicitly, it is required to compute the value of the definite integral,

$$I = \int_a^b y dx$$

Let the interval [a,b] be divided into n equal sub-intervals such that $a = x_0 < x_1 < x_2 < \cdots < x_n = b$. Clearly, $x_n = x_0 + nh$. Hence the integral becomes,

$$I = \int_{x_0}^{x_n} y dx$$

Approximating y by Newton's Forward Difference formula, we obtain,

$$I = \int_{x_0}^{x_n}\left[y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots\right]dx$$

Since, $x = x_0 + ph, dx = hdp$ and hence the above integral becomes,

$$I = h\int_0^n\left[y_0 + p\Delta y_0 + \frac{p(p-1)}{2}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{6}\Delta^3 y_0 + \cdots\right]dp$$

which gives on simplification,

$$\int_{x_0}^{x_n} y dx = nh\left[y_0 + \frac{n}{2}\Delta y_0 + \frac{n(2n-3)}{12}\Delta^2 y_0 + \frac{n(n-2)^2}{24}\Delta^3 y_0 + \cdots\right]$$

From this general formula, we can obtain different integration formulae by putting n=1,2,3,…etc.

Trapezoidal Rule

Setting n=1 in the general formula, all differences higher than the first will become zero and we obtain,

$$\int_{x_0}^{x_1} y dx = h\left(y_0 + \frac{1}{2}\Delta y_0\right) = \frac{h}{2}(y_0 + y_1)$$

For the next interval $[x_1, x_2]$, we deduce similarly,

$$\int_{x_1}^{x_2} ydx = \frac{h}{2}(y_1 + y_2)$$

and so on. For the last interval $[x_{n-1}, x_n]$, we have,

$$\int_{x_{n-1}}^{x_n} ydx = \frac{h}{2}(y_{n-1} + y_n)$$

Combining all these expressions, we obtain the rule,

$$\int_{x_0}^{x_n} ydx = \frac{h}{2}[y_0 + 2(y_1 + y_2 + \cdots + y_{n-1}) + y_n]$$

which is known as the *trapezoidal rule*.

Simpson's 1/3-Rule

Setting n=2 in the general formula, all differences higher than the second will become zero and we obtain,

$$\int_{x_0}^{x_2} ydx = 2h\left(y_0 + \Delta y_0 + \frac{1}{6}\Delta^2 y_0\right) = \frac{h}{3}(y_0 + 4y_1 + y_2)$$

Similarly,

$$\int_{x_2}^{x_4} ydx = \frac{h}{3}(y_2 + 4y_3 + y_4)$$

and finally,

$$\int_{x_{n-2}}^{x_n} ydx = \frac{h}{3}(y_{n-2} + 4y_{n-1} + y_n)$$

Summing up, we obtain,

$$\int_{x_0}^{x_n} ydx = \frac{h}{3}[y_0 + 4(y_1 + y_3 + y_5 + \cdots) + 2(y_2 + y_4 + y_6 + \cdots) + y_n]$$

which is known as the *Simpson's 1/3 rule*. It should be noted that this rule requires the division of the whole range into an even number of sub-intervals of width h.

Simpson's 3/8-Rule

Setting n=3 in the general formula, we observe that all the differences higher than the third will become zero and we obtain,

$$\int_{x_0}^{x_3} ydx = 3h\left(y_0 + \frac{3}{2}\Delta y_0 + \frac{3}{4}\Delta^2 y_0 + \frac{1}{8}\Delta^3 y_0\right) = \frac{3h}{8}(y_0 + 3y_1 + 3y_2 + y_3)$$

Similarly,

$$\int_{x_3}^{x_6} ydx = \frac{3h}{8}(y_3 + 3y_4 + 3y_5 + y_6)$$

and so on. Summing up all these, we obtain,

$$\int_{x_0}^{x_n} ydx = [y_0 + 3(y_1 + y_2 + y_4 + y_5 + \cdots) + 2(y_3 + y_6 + \cdots) + y_n]$$

This is the *Simpson's 3/8 rule* but it is not as accurate as Simpson's 1/3 rule.

Double Integration [EDIT]

Numerical Solution of Ordinary Differential Equation

Taylor's Series

Consider the differential equation,

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

with the initial condition,

$$y(x_0) = y_0$$

If y(x) is the exact solution of the equation above, then Taylor's series for y(x) around $x = x_0$ is given by,

$$y(x) = y_0 + (x - x_0)y_0' + \frac{(x - x_0)^2}{2!}y_0'' + \cdots$$

If the values of y_0', y_0'', ... are known, then the equation gives a power series for y. Using the formula for total derivatives, we can write,

$$y'' = f' = f_x + y'f_y = f_x + ff_y$$

where, the suffixes denote partial derivatives with respect to the variable concerned. Similarly, we obtain,

$$y''' = f'' = f_{xx} + f_{xy}f + f\left(f_{yx} + f_{yy}f\right) + f_y\left(f_x + f_yf\right)$$
$$= f_{xx} + 2ff_{xy} + f^2 f_{xy} + f_xf_y + ff_y^2$$

and other higher derivatives of y. The method can easily be extended to simultaneous and higher order differential equations.

Euler's Method

Suppose that we wish to solve $y' = f(x, y)$ for values of y at $x = x_r = x_0 + rh$ $(r = 1,2, \ldots)$. Integrating the equation, we get,

$$y_1 = y_0 + \int_{x_0}^{x_1} f(x, y)dx$$

Assuming that $f(x, y) = f(x_0, y_0)\ in\ x_0 \leq x \leq x_1$, this gives Euler's formula

$$y_1 = y_0 + hf(x_0, y_0)$$

Similarly, for the range $x_0 \leq x \leq x_1$, we have,

$$y_2 = y_1 + \int_{x_1}^{x_2} f(x, y)dx$$

Substituting $f(x_1, y_1)$ for $f(x, y)$ in $x_0 \leq x \leq x_1$, we obtain,

$$y_2 = y_1 + hf(x_1, y_1)$$

Proceeding in this way, we obtain the general formula,

$$y_{n+1} = y_n + hf(x_n, y_n), \quad n = 0,1,2, \ldots$$

The process is very slow and to obtain reasonable accuracy with Euler's method, we need to take a smaller value for h, Because of this restriction on h, the method is unusable for practical use.

Modified Euler's Method [EDIT]

Suppose that we wish to solve $y' = f(x, y)$ for values of y at $x = x_r = x_0 + rh$ $(r = 1,2, \ldots)$. Integrating the equation, we get,

$$y_1 = y_0 + \int_{x_0}^{x_1} f(x, y)dx$$

Approximating the integral given in the equation above by means of Trapezoidal rule to obtain,

$$y_1 = y_0 + \frac{h}{2}[f(x_0, y_0) + f(x_1, y_1)]$$

We thus obtain the iteration formula,

Runge-Kutta Method

Suppose that we wish to solve $y' = f(x, y)$ for values of y at $x = x_r = x_0 + rh$ $(r = 1,2, \ldots)$. Integrating the equation, we get,

$$y_1 = y_0 + \int_{x_0}^{x_1} f(x, y)dx$$

Approximating the integral given in the equation above by means of Trapezoidal rule to obtain,

$$y_1 = y_0 + \frac{h}{2}[f(x_0, y_0) + f(x_1, y_1)]$$

Substituting $y_1 = y_0 + hf(x_0, y_0)$ on the right side of the equation above, we obtain,

$$y_1 = y_0 + \frac{h}{2}[f_0 + f(x_0 + h, y_0 + hf_0)] \qquad where, f_0 = f(x_0, y_0)$$

If now we set, $k_1 = hf_0$ $\quad and \quad$ $k_2 = hf(x_0 + h, y_0 + k_1)$ then the above equation becomes,

$$y_1 = y_0 + \frac{1}{2}(k_1 + k_2)$$

This is the **second-order Runge-Kutta** formula.

The **forth-order Runge-Kutta** formula is as follows-

$$y_1 = y_0 + \frac{1}{6}(k_1 + k_2 + k_3 + k_4)$$

where,

$$k_1 = hf(x_0, y_0)$$
$$k_2 = hf\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_1\right)$$
$$k_3 = hf\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_2\right)$$
$$k_4 = hf(x_0 + h, y_0 + k_3)$$

Matrices and Linear Systems of Equations

Solution of Linear Systems-Direct Methods

Matrix Inversion [EDIT]

Gauss Elimination Method

This is the elementary elimination method and it reduces the system of equations to an equivalent upper-triangular system, which can be solved by back substitution.
Let's consider the system of n linear equations in n equations,

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2$$
$$\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots$$
$$a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n = b_n$$

There are two steps in the solution of the system-

1. The elimination of unknowns
2. Back substitution

Step-1:
The unknowns are eliminated to obtain an upper-triangular system. To eliminate x_1 from the second equation, we multiply the first equation by $-\frac{a_{21}}{a_{11}}$ and obtain,

$$-a_{21}x_1 - a_{12}\frac{a_{21}}{a_{11}}x_2 - a_{13}\frac{a_{21}}{a_{11}}x_3 - \cdots - a_{1n}\frac{a_{21}}{a_{11}}x_n = -b_1\frac{a_{21}}{a_{11}}$$

Adding the above equation to the second equation, we obtain,

$$\left(a_{22} - a_{12}\frac{a_{21}}{a_{11}}\right)x_2 + \left(a_{23} - a_{13}\frac{a_{21}}{a_{11}}\right)x_3 + \cdots + \left(a_{2n} - a_{1n}\frac{a_{21}}{a_{11}}\right)x_n = b_2 - b_1\frac{a_{21}}{a_{11}}$$

Which can be written as,

$$a'_{22}x_2 + a'_{23}x_3 + \cdots + a'_{2n}x_n = b'_2$$

Similarly, we can multiply the first equation by $-\frac{a_{31}}{a_{11}}$ and add it to the third equation of the system. This eliminates x_1 from the third equation and we obtain,

$$a'_{32}x_2 + a'_{33}x_3 + \cdots + a'_{3n}x_n = b'_3$$

In a similar fashion, we can eliminate x_1 from the remaining equations of the system and obtain the new system,

$$\begin{gathered} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a'_{22}x_2 + a'_{23}x_3 + \cdots + a'_{2n}x_n = b'_2 \\ a'_{32}x_2 + a'_{33}x_3 + \cdots + a'_{3n}x_n = b'_3 \\ \cdots \\ \cdots \\ a'_{n2}x_2 + a'_{n3}x_3 + \cdots + a'_{nn}x_n = b'_n \end{gathered}$$

Next, we eliminate x_2 from the last (n-2) equations of the new system. Now, to eliminate x_2 from the third equation of the new system, we multiply the equation by $-\frac{a'_{32}}{a'_{22}}$ and add it to the third equation. Repeating this process with the remaining equations, we obtain another system,

$$\begin{gathered} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a'_{22}x_2 + a'_{23}x_3 + \cdots + a'_{2n}x_n = b'_2 \\ a''_{33}x_3 + \cdots + a''_{3n}x_n = b''_3 \\ \cdots \\ \cdots \\ a''_{n3}x_3 + \cdots + a''_{nn}x_n = b''_n \end{gathered}$$

It is easily seen that this procedure can be continued to eliminate x_3 from the fourth equation onwards, x_4 from the fifth equation onwards etc. till we finally find the upper triangular form:

$$\begin{gathered} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a'_{22}x_2 + a'_{23}x_3 + \cdots + a'_{2n}x_n = b'_2 \\ a''_{33}x_3 + \cdots + a''_{3n}x_n = b''_3 \\ \cdots \\ \cdots \\ a^{n-1}_{nn}x_n = b^{n-1}_n \end{gathered}$$

where a^{n-1}_{nn} indicates that the element a_{nn} has changed $(n-1)$ times.

Step-2:

From the last equation of this system, we obtain,

$$x_n = \frac{b^{n-1}_n}{a^{n-1}_{nn}}$$

This is then substituted in the $(n-1)th$ equation to x_{n-1} and the process is repeated to compute the other unknowns. We have therefore first computed x_n, then $x_{n-1}, x_{n-2}, \ldots, x_2, x_1$ in that order. Due this reason, the process is called **back substitution**.

It is important to notice that in the process of obtaining the above system, we have multiplied the first row by $-\frac{a_{21}}{a_{11}}$, i.e. we have divided it by a_{11} which is assumed to be non-zero. For this reason, the first equation in the first new system is called the **pivot equation** and a_{11} is called the **pivot** or **pivotal element**. This method obviously fails if $a_{11} = 0$ or very close to 0. If the pivot is 0, the entire process fails and if it is close to 0, round-off errors may occur. These problems can be avoided by pivoting.

If a_{11} is 0 or very small compared to the other co − efficients of the equation, then we find the largest available coefficient in the columns below the pivot equation and then interchange the two rows. In this way, we obtain a new pivot equation with a non-zero pivot. Such process is called **partial pivoting**, since in this case, we search only the columns for the largest element. If, on the other hand, we search both columns and rows for the largest element, the procedure is called **complete pivoting**.

It is obvious that complete pivoting involves more complexity in computations since interchange of columns means change of order of unknowns which invariably requires more programming effort. In comparison, partial pivoting is easily adopted to programming. Due to this reason, complete pivoting is rarely used.

Gauss-Jordan Method

This is a modification of the Gauss elimination method; the essential difference being that when an unknown is eliminated it is eliminated from all equations. The method does not require back substitution.

Given a system,

$$a_{11}x + a_{12}y + a_{13}z = b_1$$
$$a_{21}x + a_{22}y + a_{23}z = b_2$$
$$a_{31}x + a_{32}y + a_{33}z = b_3$$

The augmented matrix of the system is,

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

Now we have to perform row and column operation on the matrix and obtain its normal form. Say, its normal form is,

$$\begin{bmatrix} 1 & 0 & 0 & b_1' \\ 0 & 1 & 0 & b_2' \\ 0 & 0 & 1 & b_3' \end{bmatrix}$$

Hence, the root will be $x = b_1', \quad y = b_2', \quad z = b_3'$

To apply this method to solve a system, the system has to be consistent, i.e. the rank of the augmented matrix has to be equal to the number of variables in the system.

Difference between Gauss elimination and Gauss Jordan method

Gaussian elimination and Gauss-Jordan elimination are both used to solve systems of linear equations, as well as finding inverses of non-singular matrices. If, using elementary row operations, the augmented matrix is reduced to row echelon form (REF), then the process is called Gaussian elimination. If the matrix is reduced to reduced row echelon form (RREF), the process is called Gauss-Jordan elimination. In the case of Gaussian elimination, assuming that the system is consistent, the solution set can be obtained by "back-substitution" whereas, if the matrix is in reduced row echelon form, the solution set can be obtained "directly" from the final matrix.

The main difference is that Gaussian elimination brings the augmented matrix into the "lower triangular form" on the left side, but Gauss-Jordan makes the augmented matrix an "identity" matrix on the left.

LU Decomposition of a Matrix

A square matrix A can be factorized into the form LU, where L is unit lower triangular and U is upper triangular, if all the principle minors of Anon-singular, i.e. if,

$$a_{11} \neq 0 \qquad \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \neq 0 \qquad \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \neq 0, \qquad etc$$

It is a standard result of linear algebra that such a factorization, when it exists, is unique.

Let's consider the following system-

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2$$
$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3$$

which can be written in the form, $\boldsymbol{AX} = \boldsymbol{B} \ldots\ldots\ldots\ldots\ldots\ldots (1)$

Let, $\mathrm{A} = \mathrm{LU}$, where,

$$L = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}$$

and $$U = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & 0 \end{bmatrix}$$

Hence, eq. (1) becomes, $LUX = B$

Since $A = LU$, we can write

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & 0 \end{bmatrix} = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

Multiplying the matrices on the left and equating the corresponding elements of both sides, we obtain,

$$\boldsymbol{u_{11} = a_{11}} \qquad \boldsymbol{u_{12} = a_{12}} \qquad \boldsymbol{u_{13} = a_{13}}$$

$$\boldsymbol{l_{21}u_{11} = a_{21}}$$
$$\Rightarrow l_{21} = \frac{a_{21}}{u_{11}} = \frac{a_{21}}{a_{11}}$$

$$\boldsymbol{l_{21}u_{12} + u_{22} = a_{22}}$$
$$\Rightarrow u_{22} = a_{22} - l_{21}u_{12}$$
$$\Rightarrow u_{22} = a_{22} - \frac{a_{21}}{a_{11}}a_{12}$$

$$\boldsymbol{l_{21}u_{13} + u_{23} = a_{23}}$$
$$\Rightarrow u_{23} = a_{23} - l_{21}u_{13}$$
$$\Rightarrow u_{23} = a_{23} - \frac{a_{21}}{a_{11}}a_{13}$$

$$\boldsymbol{l_{31}u_{11} = a_{31}}$$
$$\Rightarrow l_{31} = \frac{a_{31}}{u_{11}} = \frac{a_{31}}{a_{11}}$$

$$\boldsymbol{l_{31}u_{12} + l_{32}u_{22} = a_{32}}$$
$$\Rightarrow l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}}$$
$$\Rightarrow l_{32} = \frac{a_{32} - \frac{a_{31}}{a_{11}}a_{12}}{a_{22} - \frac{a_{21}}{a_{11}}a_{12}}$$
$$\Rightarrow l_{32} = \frac{a_{11}a_{32} - a_{12}a_{31}}{a_{11}a_{22} - a_{12}a_{21}}$$

$$\boldsymbol{l_{31}u_{13} + l_{32}u_{23} + u_{33} = a_{33}}$$
$$\Rightarrow u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$
$$\Rightarrow u_{33}$$
$$= a_{33} - \frac{a_{31}}{a_{11}}a_{13}$$
$$- \left(\frac{a_{11}a_{32} - a_{12}a_{31}}{a_{11}a_{22} - a_{12}a_{21}}\right)\left(a_{23} - \frac{a_{21}}{a_{11}}a_{13}\right)$$

LU Decomposition from Gauss Elimination [EDIT]

Solution of Linear Systems-Iterative Methods [EDIT]

Jacobi's Method [EDIT]

Gauss-Seidel Method [EDIT]

Numerical Solution of Partial Differential Equations [EDIT]

The general second order linear partial differential equation is of the form

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial x \partial y} + C\frac{\partial^2 u}{\partial y^2} + D\frac{\partial u}{\partial x} + E\frac{\partial u}{\partial y} + Fu = G$$

Which can be written as,

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G$$

Where A, B, C, …., G are all functions of x and y.

Boundary-value and Eigen value Problems [EDIT]

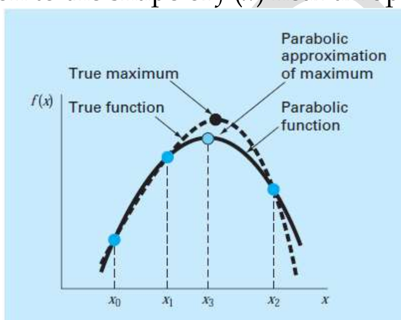
General methods for Boundary-value problems

Eigen value problem

One Dimensional Unconstrained Optimization [EDIT]

Parabolic Interpolation

Parabolic interpolation takes advantage of the fact that a second-order polynomial often provides a good approximation to the shape of $f(x)$ near an optimum (the figure below).



Just as there is only one straight line connecting two points, there is only one quadratic polynomial or parabola connecting three points. Thus, if we have three points that jointly bracket an optimum, we can fit a parabola to the points. Then we can differentiate it, set the result equal to zero, and solve for an estimate of the optimal x. It can be shown through some algebraic manipulations that the result is

$$x_3 = \frac{f(x_0)(x_1^2 - x_2^2) + f(x_1)(x_2^2 - x_0^2) + f(x_2)(x_0^2 - x_1^2)}{2f(x_0)(x_1 - x_2) + 2f(x_1)(x_2 - x_0) + 2f(x_2)(x_0 - x_1)}$$

Where x_0, x_1 and x_2 are the initial guesses, and x_3 is the value of x that corresponds to the maximum value of the parabolic fit to the guesses. After generating the new point, there are two strategies for selecting the points for the next iteration. The simplest approach, which is similar to the secant method, is to merely assign the new points sequentially. That is, for the new iteration $z_0 = z_1, z_1 = z_2$ and $z_2 = z_3$.

To understand better, spectate example 13.2 from Numerical Methods for Engineers by Steven C. Chapra, page-360.

[References:

1. Class Lecture of SA mam
2. Numerical Methods for Engineers_ Steven C. Chapra
3. Introductory Methods of Numerical Analysis – S.S. Sastry
4. Wikipedia]